Mathematical Foundations for ML

Lecture 1. Probability Basics & Binary Indicator
- Basic Probability Calculus
- Expectation & Variance
- Sum of Random Variables

Lecture 2. Discrete Distributions & Classification
- Common Discrete Distributions
- Optimal Binary Classification
- Classification Error Estimation
- Nearest Neighbor Classification
- Histogram Classifier
- "Plug-in" Classifier
- Markov's Inequality
- Jensen's Inequality

Lecture 3. Multivariate Gaussian Models
- Multivariate Gaussian (or Normal, MVN)
- MVN Multi-class Classification

Lecture 4. Learning MVN Classifiers
- "Plug-in" MVN Classifier
- Analysis of Probability of Error
- Empirical Mean & Covariance

Lecture 5. Likelihood & Kullback-Leibler Divergence
- Binary MVN Classification

Lecture 6. Maximum Likelihood Estimation
- Maximum Likelihood Estimate (MLE)
- MLE and KL-Divergence

Lecture 7. Sufficient Statistics
- Definition of Sufficient Statistics
- Fisher-Neyman Factorization
- Rao-Blackwell Theorem

Lecture 8. Asymptotic Analysis of MLE
- Convergence of Log-Likelihood to KL
- Asymptotic Distribution of MLE
- Review of Central Limit Theorem

Lecture 9. Generalized Linear Models
- Linear Modeling Approach
- The Exponential Family Models
- Generalized Linear Model Examples

Lecture 10. Linear Models Optimization
- Common Loss Functions
- Optimization Approaches

Lecture 11. Gradient Descent
- Gradient Descent & Strong Convexity
- Stochastic Gradient Descent
- Subgradients for Non-differentiable $f$

Lecture 12. Analysis of Stochastic Gradient Descent
- General SGD Iteration Analysis

Lecture 13. Bayesian Inference
- Bayesian Inference Components
- Maximum a Posteriori Estimator (MAP)
- Bias-Variance Decomposition of MSE
- MVN in Bayesian Inference
- Bayesian Linear Modeling

Lecture 14. Proximal Gradient Algorithms
- Proximal Operator & Soft-Thresholding
- Special Case of Squared Error Loss
- General Proximal Gradient Algorithm

Lecture 15. Analysis of Soft-Thresholding
- Lasso Regression Soft-Thresholding Estimator

Lecture 16. Concentration Inequalities
- Central Limit Theorem
Lecture 1. Probability Basics & Binary Indicator

Basic Probability Calculus

Let \( X, Y \) be Random Variables (RVs).

- **joint probability**: \( p(x, y) \)
- **Marginal probability**: \( p(x) = \sum_y p(x, y) \) if discrete or \( \int_y p(x, y) \) if continuous
- **Conditional probability**: \( p(y|x) \)
  - \( p(x, y) = p(y|x)p(x) \)
  - If \( X, Y \) are independent, then \( p(x, y) = p(x)p(y) \)

Expectation & Variance

Assume discrete random variables.

- **Expectation of \( X \)**: \( \mathbb{E}[X] = \sum_x xp(x) \)
  - \( \mathbb{E}[f(X)] = \sum_x f(x)p(x) \)
  - \( \mathbb{E}[X + Y] = \mathbb{E}[X] + \mathbb{E}[Y] \)
  - \( \mathbb{E}[XY] = \sum_x \sum_y xp(x, y) \) is not a function of \( \mathbb{E}[X] \) and \( \mathbb{E}[Y] \) in general
  - If \( X, Y \) are independent, then \( \mathbb{E}[XY] = \mathbb{E}[X]\mathbb{E}[Y] \)
- **Conditional expectation**: \( \mathbb{E}[Y|X = x] = \sum_y y p(y|x) \)
  - If \( X, Y \) are independent, then \( \mathbb{E}[Y|X = x] = \mathbb{E}[Y] \)
- **Variance of \( X \)**: \( \text{Var}(X) = \mathbb{E}[(X - \mathbb{E}[X])^2] \)
Sum of Random Variables

Let $X_i$ be random variables.

- $\mathbb{E}[\sum_{i=1}^{n} X_i] = \sum_{i=1}^{n} \mathbb{E}[X_i]$  
- $\text{Var}(\sum_{i=1}^{n} X_i) = \mathbb{E}[(\sum_{i=1}^{n} (X_i - \mathbb{E}[X_i])^2] = \sum_{i=1}^{n} \text{Var}(X_i) + 2 \sum_{i<j} \text{Cov}(X_i, X_j)$
  - $\text{Cov}(X_i, X_j)$ is the covariance between $X_i, X_j$

If the $X_i$’s are independent, $\text{Var}(\sum_{i=1}^{n} X_i) = \sum_{i=1}^{n} \text{Var}(X_i)$

Let $\mathbb{I}_x$ be the binary indicator variable of event $x$.

- The empirical probability of event $x$ happening is $\hat{p}_x = \frac{1}{n} \sum_{i=1}^{n} \mathbb{I}_x$
- $\mathbb{E}[\hat{p}_x] = \frac{1}{n} \sum_{i=1}^{n} \mathbb{E}[\mathbb{I}_x] = \frac{1}{n} \sum_{i=1}^{n} p_x = p_x$
  - Meaning the empirical probability is an unbiased estimator of the true probability
- $\text{Var}(\hat{p}_x) = \mathbb{E}[(\hat{p}_x - p_x)^2] = \frac{1}{n^2} \sum_{i=1}^{n} \mathbb{E}[\mathbb{I}_x - p_x]^2 = \frac{p_x(1-p_x)}{n}$
  - Meaning if sample size $n$ is small, however, it may have a large variance

Lecture 2. Discrete Distributions & Classification

Common Discrete Distributions

Common discrete probability distributions:

- **Bernoulli**: binary variable $X$ taking value 1 with probability $p$
  - $p(x) = p^x(1-p)^{1-x}$
  - $\mathbb{E}[X] = p$
  - $\text{Var}(X) = p(1-p)$

- **Binomial**: consider $n$ independent and identically distributed (i.i.d.) Bernoulli variables $X_i$
  - $p(x_1, \ldots, x_n) = \prod_{i=1}^{n} p(X_i = x_i) = \prod_{i=1}^{n} p^x_i(1-p)^{1-x_i}$
  - The sum of i.i.d. Bernoullis $S_n = \sum_{i=1}^{n} X_i$ follows a Binomial distribution:
    - $p(S_n = k) = \binom{n}{k} p^k(1-p)^{n-k} = \frac{n!}{k!(n-k)!} p^k(1-p)^{n-k}$
    - $\mathbb{E}[S_n] = np$
    - $\text{Var}(S_n) = np(1-p)$

- **Multinomial**: consider $n$ i.i.d. random variables $X_i$ that take values in $\{a_1, \ldots, a_m\}$
  - $p(x_1, \ldots, x_n) = \prod_{i=1}^{n} \prod_{j=1}^{m} \mathbb{I}_{x_i = a_j}$
  - Let $K_j$ be the number of times value $a_j$ appears, $K_j$ follows a Multinomial distribution:
    - $p(k_1, \ldots, k_m) = \binom{n}{k_1, \ldots, k_m} \prod_{j=1}^{m} p_j^{k_j} = \frac{n!}{k_1! \cdots k_m!} \prod_{j=1}^{m} p_j^{k_j}$
    - $\mathbb{E}[K_j] = np_j$
    - $\text{Var}(K_j) = np_j(1-p_j)$

- **Poisson**: non-negative integer-valued variable $X$ with distribution:
  - $p(X = k) = \frac{\lambda^k e^{-\lambda}}{k!}, \lambda > 0$
  - $\mathbb{E}[X] = \lambda$
  - $\text{Var}(X) = \lambda$

Optimal Binary Classification

The goal of classification is to learn a mapping $f$ from the feature space $\mathcal{X}$ to the label space $\mathcal{Y}$.

- The mapping $f$ is called a classifier
  - Assume $\mathcal{X} = \mathbb{R}^d$ and $\mathcal{Y} = \{0, 1\}$ in the following examples
- We measure the error of a classifier using a loss function $L$
  - e.g., the 0-1 loss $L(f(x), y) = \mathbb{I}_{f(x) \neq y}$
- The risk is defined to be the expectation of the loss: $R(f) = \mathbb{E}[L(f(X), Y)]$
  - In the 0-1 loss case, $R(f) = p(f(X) \neq Y)$
  - In the 0-1 loss case, the total number of mistakes $m$ is a binomially distributed RV
Performance of a classifier can be evaluated in terms of how close its risk is to the Bayes risk.

- The Bayes risk $R^* = \inf_f R(f)$
- The Bayes classifier achieves the Bayes risk

$$f^*(x) = \begin{cases} 1, & \eta(x) \geq \frac{1}{2} \\ 0, & \text{otherwise} \end{cases}$$

where $\eta(x) = p(Y = 1 | X = x)$. We have $R(f^*) = R^*$.

- Probability of error of the optimal classifier

$$p(f^*(X) \neq Y) = \mathbb{E}[\mathbb{I}_{\{f^*(x) \neq Y\}}] = \mathbb{E}[\mathbb{E}[\mathbb{I}_{\{f^*(X) \neq Y\}} | X]] = \mathbb{E}[\min(\eta(X), 1 - \eta(X))]$$

**Classification Error Estimation**

A common approach to estimate the error rate of classifier $f$ is to evaluate on a test set $\{X_i, Y_i\}_{i=1}^n$ drawn i.i.d. from $\mathbb{P}_{XY}$.

- The empirical error rate is $\hat{p}_f = \frac{1}{n} \sum_{i=1}^n \mathbb{I}_{\{f(X_i) \neq Y_i\}}$
  - $\hat{p}_f$ has a Binomial distribution

- $\mathbb{E}[\hat{p}_f] = p_f$
- $\mathbb{E}[-\hat{p}_f] = \frac{n(1-p_f)}{n}$

**Nearest Neighbor Classification**

The nearest neighbor classifier labels a new point $X$ by finding the closest point in the training set and assigning the corresponding label of it.

- The distance function can be any valid distance measure:
  - e.g., the Euclidean distance $\text{dist}(X, X_i) = ||X - X_i||_2$
  - $\lim_{n \to \infty} p(f^{NN}(X) \neq Y) = \mathbb{E}[2\eta(X)(1 - \eta(X))]$, denoted as $R_{\infty}^{NN}$, is the asymptotic error
  - $R_{\infty}^{NN} \leq 2R^*$

**Histogram Classifier**

The histogram classifier is based on a partitioning of a hypercube space into $M$ smaller cubes of "bins" of equal size. Let the bins be denoted $\{B_j\}_{j=1}^M$, the classifier is an assignment of 0 or 1 to each bin:

- A reasonable rule is to assign the majority vote of training examples that fall into each bin
  - i.e., if $\hat{p}_j = \frac{\sum_{i=1}^n \mathbb{I}_{\{B_j \cap X_i\}}}{\sum_{i=1}^n \mathbb{I}_{(X_i \in B_j)}} \geq \frac{1}{2}$, label 1, otherwise label 0
  - Equivalently, we can have an estimator $\tilde{\eta}_n(x) = \sum_{j=1}^M \hat{p}_j \mathbb{I}_{\{x \in B_j\}}$
  - and classify according to if $\tilde{\eta}_n(x) \geq \frac{1}{2}$ or not; label 1 if $\geq \frac{1}{2}$
  - The bias of histogram classifier tends to 0 as $M \to \infty$; the variance tends to 0 as $n \to 0$
  - We say histogram classifiers are universally consistent, i.e., their error rate converges to the Bayes error rate

"Plug-in" Classifier

Let $\hat{\eta}$ be any approximation to $\eta$, the "plug-in" classifier is:

$$f(x) = \begin{cases} 1, & \hat{\eta}(x) \geq \frac{1}{2} \\ 0, & \text{otherwise} \end{cases}$$

- $p(f(X) \neq Y) - p(f^*(X) \neq Y) \leq 2\mathbb{E}[||\eta(X) - \hat{\eta}(X)||]$  

**Markov’s Inequality**

Let $X$ be a nonnegative random variable, the Markov’s inequality states that $p(X \geq t) \leq \frac{\mathbb{E}[X]}{t}$.

**Jensen’s Inequality**

For any convex function $\varphi$, that is, $\varphi(\lambda x_1 + (1 - \lambda)x_2) \leq \lambda \varphi(x_1) + (1 - \lambda)\varphi(x_2)$ for any $\lambda \in [0, 1]$, we have Jensen’s inequality: $\mathbb{E}[\varphi(X)] \geq \varphi(\mathbb{E}[X])$.

- Obvious results: $\mathbb{E}[X^2] \geq (\mathbb{E}[X])^2$ and $\mathbb{E}[|X|^3] \geq (\mathbb{E}[|X|])^3$
Multivariate Gaussian (or Normal, MVN)

Let the feature space be $\mathbb{R}^d$, the MVN density function is given by:

$$p(x) = \frac{1}{\sqrt{(2\pi)^d |\Sigma|}} e^{-\frac{1}{2}(x-\mu)^T \Sigma^{-1} (x-\mu)}$$

where $\mu$ is the mean and $\Sigma = \mathbb{E}[(x - \mu)(x - \mu)^T]$ is the covariance matrix. We write $x \sim \mathcal{N}(\mu, \Sigma)$.

- $\Sigma$ is always positive-semi-definite in order to be a valid covariance matrix
- Linear transformations of Gaussian random variables are also Gaussian
  - $Ax + b \sim \mathcal{N}(A\mu + b, A\Sigma A^T)$

**MVN Multi-class Classification**

Consider features $x$ of examples belonging to class $j$, i.e., the class conditional distributions of $x$, are all Gaussian:

$$x|y = j \sim \mathcal{N}(\mu_j, \Sigma_j)$$

- The optimal classification rule is $\hat{y}(x) = \arg \max \ p(y = j|x)$
- By Bayes rule, $p(y = j|x) = \frac{p(x|y = j)p(y = j)}{p(x)}$
  - $p(y = j)$ is the marginal probability that a random example belongs to class $j$, often called the prior probability of class $j$
  - $p(x)$ is the marginal density of $x$ for classification of a given $x$, this value is a constant
  - Therefore, the rule can be expressed as $\hat{y}(x) = \arg \max \ p(y = j)p(x|y = j)$
  - $p(x|y = j)$ is called class conditional likelihood of $x$
- Consider the special case of binary classification:

$$\hat{y}(x) = \begin{cases} 1, & \frac{p(y=1|x)}{p(y=0|x)} > \frac{p(y=0|x)}{p(y=1|x)} \\ 0, & \text{otherwise} \end{cases}$$

This is called the log-likelihood ratio test (LRT).

- For Gaussian class-conditional densities, the ratio is a quadratic function in $x$, so the decision boundary is a quadratic curve/surface in the feature space
- For Gaussian class-conditional densities with equal covariances AND equal prior probabilities, the ratio simplifies to:

$$\hat{y}(x) = \begin{cases} 1, & 2(\mu_1 - \mu_0)^T \Sigma^{-1} x \geq \mu_1^T \Sigma^{-1} \mu_1 - \mu_0^T \Sigma^{-1} \mu_0 \\ 0, & \text{otherwise} \end{cases}$$

which is a linear classifier (Fisher’s linear discriminant).

- $\frac{p(y=0)}{p(y=1)} = \gamma > 0$ is the threshold of the test; LRT, with an appropriate threshold, is optimal

**Lecture 4. Learning MVN Classifiers**

"Plug-in" MVN Classifier

Consider a set of training data. Denote training data with label $j$ as $\{x_i\}_{i:y_i = j}$

- $\mu_j = \frac{1}{n_j} \sum_{i:y_i = j} x_i$
- $\Sigma_j = \frac{1}{n_j} \sum_{i:y_i = j} (x_i - \mu_j)(x_i - \mu_j)^T$

If we “plug-in” these estimates to obtain an MVN model for class $j$, i.e., let $x|y = j \sim \mathcal{N}(\mu_j, \Sigma_j)$ for all classes, we obtain a "plug-in" MVN classifier.

**Analysis of Probability of Error**

Consider the simple setting $x|y = +1 \sim \mathcal{N}(\theta, I)$ and $x|y = -1 \sim \mathcal{N}(-\theta, I)$.

- The optimal classification rule after applying LRT is:
This achieves the minimum probability of error
\[ p(f^*(x) \neq y) = p(x^T \theta > 0 | y = 1) + p(x^T \theta < 0 | y = 0) \]
(due to symmetry of the problem)

Note that \( x^T \theta y = -1 \sim \mathcal{N}(0, ||\theta||^2) \), so the probability of error is equal to the probability that an RV \( x \sim \mathcal{N}(0, ||\theta||^2) \) exceeds \( ||\theta||^2 \)

Apply Markov's inequality,
\[ p(\theta > ||\theta||^2) \leq p(\theta^2 > ||\theta||^4) \leq \frac{E[\theta^2]}{||\theta||^4} = \frac{1}{||\theta||^2} \]

Insight: the probability of error decreases as the distance between the means increases

Now consider a learning setup: we don't know the value of \( \theta \) but we have an estimator from training samples \( \hat{\theta} = \frac{1}{n} \sum_{i=1}^{n} y_i x_i \).

- Plug-in the estimate, the probability of error is \( p(\hat{f}(x) \neq y) = p(x^T \hat{\theta} > 0 | y = 1) \)
  
  Since both \( x \) and \( \hat{\theta} \) are RVs, \( x^T \hat{\theta} \) does not have a simple distribution

Decomposing into an offset + a zero-mean component: let \( x = \theta + e_1 \) and \( \hat{\theta} = \theta + e_2 \), where \( e_1 \sim \mathcal{N}(0, I) \) and \( e_2 \sim \mathcal{N}(0, \frac{1}{\lambda} I) \)

Expand \( x^T \hat{\theta} \) and apply Markov's inequality, eventually all cross-terms vanish; taking the expectation first w.r.t. \( e_1 \) (consider \( e_2 \) as given) and then w.r.t. \( e_2 \), we have
\[ p(\hat{f}(x) \neq y) \leq \frac{(1+\frac{1}{\lambda}) \theta^2 \lambda^2}{\theta^4} \]

Notice that if \( n \gg d \), the bound is essentially equal to the one of the Bayes classifier \( \frac{1}{||\theta||^2} \)

Comparing it with histogram classifiers:

- MVN plug-in classifiers require class-conditional densities to be strong MVNs and work well if the number of samples \( n > d \)
- Histogram classifiers require nothing from data distributions but work well only if \( n > 2^d \) ⇒ the "curse of dimensionality"

### Empirical Mean & Covariance

Are they biased/unbiased estimators?

- The empirical mean \( \hat{\mu} = \frac{1}{n} \sum_{i=1}^{n} x_i \) is an unbiased estimator of \( \mu \)
- The empirical covariance \( \hat{\Sigma} = \frac{1}{n} \sum_{i=1}^{n} (x_i - \hat{\mu})(x_i - \hat{\mu})^T \) is a biased estimator of \( \Sigma \)
  
  \[ \mathbb{E}[n\hat{\Sigma}] = n\text{Var}(x_i) = n\Sigma - \Sigma = (n-1)\Sigma \]
  so \( \mathbb{E}[\hat{\Sigma}] = \frac{n-1}{n} \Sigma \)
  
  As \( n \to \infty \), \( \hat{\Sigma} \) is an asymptotically unbiased estimator of \( \Sigma \)

#### Lecture 5. Likelihood & Kullback-Leibler Divergence

### Binary MVN Classification

Recall that binary MVN classification with equal covariances and equal prior probabilities yields an optimal linear classifier
\[ \hat{y}(x) = 1 \text{ if } w^T x + b \geq 0 \]

- \( \mathbb{E}[w^T x + b | y = 1] = (\mu_1 - \mu_0)^T \Sigma^{-1} (\mu_1 - \mu_0) \) is the squared Mahalanobis distance between the means
  
  We can write the test statistic \( w^T x + b \) as \( (\mu_1 - \mu_0)^T \Sigma^{-1} (\mu_1 - \mu_0) + z \), where \( z \) is a zero-mean RV
  
  The optimal classifier is \( f^*(x) = 1 \text{ if } \eta(x) \geq 1, \text{ i.e. } f^*(x) = 1 \text{ if } \eta(x) \geq \frac{1}{\eta(x)} \geq 1 \)

### Kullback-Leibler (KL) Divergence

Denote the log-likelihood ratio as \( \Lambda(x) = \log \frac{p_1(x)}{p_0(x)} \). We would like to derive a metric for intrinsically describing the difficulty of the classification problem.

- Let \( q \) be the true distribution of \( x \) (which is either \( p_0 \) or \( p_1 \) in this case)
\[ \mathbb{E}[\Lambda(x)] = \int q(x) \frac{p_1(x)}{p_0(x)} \, dx \]
\[ = \int q(x) \log \frac{p(x)}{p_0(x)} \, dx - \int \frac{q(x)}{p_1(x)} \, dx \]
\[ = \mathbb{E}[\log \frac{p(x)}{p_0(x)}] - \mathbb{E}[\frac{q(x)}{p_1(x)}] \]
\[ = D(q||p_0) - D(q||p_1) \]

where \( D(q||p_0) \) is the KL-divergence of distribution \( p_0 \) from \( q \) similarly for \( p_1 \).

- KL-divergence is non-negative, provable by convexity and Jensen’s inequality
  - If \( q = p_0 \), \( D(q||p_0) = 0 \) and here \( \mathbb{E}[\Lambda(x)] = D(p_1||p_0) \geq 0 \)
  - If \( q = p_1 \), \( D(q||p_1) = 0 \) and here \( \mathbb{E}[\Lambda(x)] = -D(p_0||p_1) \leq 0 \)
- In general, KL-divergences are not symmetric: \( D(q||p) \neq D(p||q) \)

- For binary MVN classification with equal covariances, we have \( D(p_0||p_1) = \frac{1}{2} (\mu_1 - \mu_0)^T \Sigma^{-1} (\mu_1 - \mu_0) \), which is proportional to the squared Mahalanobis distance between the means
- Two classes are separable iff. the class-conditional densities do not overlap, i.e., the supports are disjoint subsets of the feature space; in this case, the integrand in KL-divergence \( D\)'s at those points is infinite, and \( D\)’s are therefore not well-defined
- \( D(p(x,y)||p(x)p(y)) \) is named the mutual information between \( x \) and \( y \)
  - If \( x \) and \( y \) are independent, this KL-divergence is 0

### Lecture 6. Maximum Likelihood Estimation

**Maximum Likelihood Estimate (MLE)**

*Maximum likelihood estimation* is a common methodology for estimating the parameters of a probabilistic model family. Its core principle is *density estimation*.

- Consider a family of probability distributions indexed by parameter(s) \( \theta \). Given a bunch of observations data \( \mathbf{x} \), we would like to make an estimate \( \hat{\theta} \) to pick the best model in the family that fits our data
  - The MLE chooses \( \hat{\theta} = \arg \max_{\theta} p(\mathbf{x}||\theta) \)
  - Viewing \( p(x||\theta) \) as a function of \( x \), it is essentially just the class-conditional density function given the model parameterized by a specific \( \theta \)
  - Viewing \( p(x||\theta) \) as a function of \( \theta \), however, we say that it is the likelihood function for different \( \theta \) values to generate the observed data \( \mathbf{x} \)
  - Suppose \( \theta \in \{0, 1\} \), i.e., binary classification, MLE in this case is equivalent to LRT
- Assume \( x_i \sim q \) for \( n \) samples and they are i.i.d., the MLE is:
  - \( \arg \max_{\theta} \prod_{i=1}^{n} p_{\theta}(x_i) \), or equivalently, \( \arg \max_{\theta} \sum_{i=1}^{n} \log p_{\theta}(x_i) \)
  - Can also express as minimization: \( \arg \min_{\theta} -\sum_{i=1}^{n} \log p_{\theta}(x_i) \)
- It is possible that the true distribution \( q \) is not a member of the parametric family under consideration

#### Examples of MLE:
- Let \( x_i \sim \text{i.i.d. Uniform}(0, \theta) \)
  - MLE \( \hat{\theta}_n = \max \{x_i\}_{i=1}^{n} \)
  - \( \text{CDF}(\hat{\theta}_n) = (\frac{1}{\theta})^{n} \)
- TODO: Structured Mean, Poisson Mean, Linear Regression

### MLE and KL-Divergence

MLE can be related with KL-divergence through the lens of loss functions:

- We can view the negative log-likelihood function as a sum of “loss functions” \(-\sum_{i=1}^{n} \log p_{\theta}(x_i) = \sum_{i=1}^{n} l_i(q, p_\theta) \)
  - \( l_i(q, p_\theta) \), or simply \( l_i(\theta) \), measures the loss incurred when using \( p_\theta \) to model \( x_i \)
  - The risk \( R(q, p_\theta) = \mathbb{E}[l_i(q, p_\theta)] = -\int q(x) \log p_\theta(x) \, dx \)
  - The excess risk \( R(q, p_\theta) - R(q, q) = \mathbb{E}[\log q(x) - \log p_\theta(x)] = D(q||p_\theta) \geq 0 \)
This shows that \( q \) minimizes the risk. Consider \( \theta^* = \arg \min_{\theta} D(q\| p_\theta) \) to be the optimal value of \( \theta \).

- If \( x \) contains multiple i.i.d. observations \( x_i \sim q \), then the MLE is \( \hat{\theta}_n = \arg \min_{\theta} -\sum_{i=1}^n \log p(x_i) \)
- By the strong law of large numbers, for any \( \theta \in \Theta, \frac{1}{n} \sum_{i=1}^n \log \frac{q(x_i)}{p(x_i)} \to D(q\| p_\theta) \) asymptotically

General technique of finding the MLE of \( \theta \):
1. Write out the likelihood function \( L \), or the log-likelihood form, or the negative form
2. Confirm that \( L \) is convex (concave). Do derivative \( L'(\theta) \) w.r.t. \( \theta \) and solve for \( L'(\theta) = 0 \)

**Lecture 7. Sufficient Statistics**

**Definition of Sufficient Statistics**
The idea is to find a lower-dimensional representation of the set of observations \( x \), denoted as \( t(x) \), that alone carries all the relevant information about model parameter \( \theta \).

- Formally, given a model family parameterized by \( \theta \) and a set of observations \( x = \{ x_i \}_{i=1}^n \)
  - Find a function \( t(x) \) that preserves all information that we can use to estimate the best \( \theta^* \)
  - \( t(x) \) is called a **sufficient statistic** of \( x \) for \( \theta \)
  - The distribution of \( x \) given \( t(x) \) is independent of \( \theta \), i.e., \( p(x|t, \theta) = p(x|t) \)
- Under MLE, the result \( \hat{\theta} \) is exclusively based on the shape of the likelihood function. Any processing/compression operations that preserve the shape will not affect the outcome of the estimation process -- this is the key idea of sufficient statistics

- Example of Bernoulli RVs where \( p(x = 1) = \theta \) and \( k = \sum_{i=1}^n x_i \) is the number of 1’s:
  - \( p(x_1, \ldots, x_n|k, \theta) = \theta^k (1-\theta)^{n-k} = \frac{1}{n!} \)
  - \( \Rightarrow k = \sum_{i=1}^n x_i \) is a sufficient statistic that carries all relevant information about \( \theta \)
  - \( k \) compresses \( \{0,1\}^n \) (n bits) to \( \{0,\ldots,n\} \) (log \( n \) bits)

A sufficient statistic is **minimal** if the dimension of \( t(x) \) cannot be further reduced while still being sufficient.

**Fisher-Neyman Factorization**
Let \( x \) be an RV with density \( p(x|\theta) \), the statistic \( t(x) \) is sufficient iff. the density can be factorized as \( p(x|\theta) = a(x) \cdot b(t(x), \theta) \) where:

- \( a(x) \) is an arbitrary function of \( x \)
- \( b(t(x), \theta) \) is a function of \( \theta \) and only depends on \( x \) through \( t(x) \)

**Proof:**
\[
p(x|t, \theta) = \frac{p(x|t, \theta)}{p(t|\theta)} = \frac{p(x|\theta)}{p(t|\theta)} = \frac{a(x)b(t, \theta)}{\int a(x')b(t, \theta)dx'} = \frac{a(x)}{\int a(x')dx'} = \text{independent of } \theta
\]

Example:

- Bernoulli: \( p(x_1, \ldots, x_n|\theta) = \theta^k (1-\theta)^{n-k} \Rightarrow k \) is sufficient for \( \theta \)
- Poisson: \( p(x_1, \ldots, x_n|\lambda) = \prod_{i=1}^n e^{-\lambda} \frac{\lambda^{x_i}}{x_i!} = (\prod_{i=1}^n \frac{1}{x_i!}) \cdot e^{-\lambda} \lambda^{x_i} \Rightarrow \sum_{i=1}^n x_i \) is sufficient for \( \lambda \)
- Gaussian: \( x_i \sim N(\mu, \Sigma) \), the pair of sample empiricals \( (\mu, \Sigma) \) is sufficient for \( (\mu, \Sigma) \)
- Uniforms: \( x_i \sim \text{Uniform}(a, b) \), \( p(x|a, b) = \frac{1}{(b-a)} \cdot 1_{a \leq x_i \leq b} \Rightarrow \min_i x_i \) and \( \max_i x_i \) are sufficient for \( a \\) and \( b \), respectively

A sufficient statistic could also be derived from the density of a joint distribution of multiple distributions that share the same parameters \( \theta \).

**Rao-Blackwell Theorem**
Assume \( x \sim p(x|\theta), \theta \in \mathbb{R} \) and \( t(x) \) is a sufficient statistic for \( \theta \). Let \( f(x) \) be an estimator of \( \theta \). The **Rao-Blackwell Theorem** considers the mean square error \( \mathbb{E}[(f(x) - \theta)^2] \):

- Define \( g(t(x)) = \mathbb{E}[f(x)|t(x)] \)
- then \( \mathbb{E}[(g(t(x)) - \theta)^2] \leq \mathbb{E}[(f(x) - \theta)^2] \), with equality iff. \( f(x) = g(t(x)) \) with probability 1, i.e., \( f \) is \( g \circ t \).

Proof through Jensen’s inequality:
Lecture 8. Asymptotic Analysis of MLE

Convergence of Log-Likelihood to KL

Consider the MLE setting:

- $\hat{\theta}_n = \arg \min_{\theta} \sum_{i=1}^{n} \log \frac{q(z_i)}{p(z_i|\theta)}$
- By the strong law of large numbers, $\frac{1}{n} \sum_{i=1}^{n} \log \frac{q(z_i)}{p(z_i|\theta)} \rightarrow D(q||p_0)$ asymptotically
- Let $\theta^* = \arg \min_{\theta} D(q||p_0)$
  - We can show that $D(q||\hat{p}_n) \rightarrow D(q||p_0)$ asymptotically

Asymptotic Distribution of MLE

Assume that the data are generated by $q = p_0$. The notation $\hat{\theta}_n \sim_{asypmp} p$ means that as $n \rightarrow \infty$, the distribution of MLE $\hat{\theta}_n$ tends to the distribution $p$.

- It is shown that $\hat{\theta}_n \sim_{asypmp} N(\theta^*, \frac{1}{n} I^{-1}(\theta^*))$.
- where $I(\theta^*)$ is the Fisher-Information Matrix (FIM), whose elements are given by $\frac{\partial^2 \log p_0}{\partial \theta \partial \theta^T}$
- This tells that the distribution tends to a Gaussian distribution. Also, $\hat{\theta}_n$ is asymptotically unbiased. The asymptotic covariance also decays, but the structure is determined by the FIM.

The FIM measures the curvature of the log-likelihood surface.
- In the case where $\theta$ is scalar, the FIM is simply the second derivative of log-likelihood
- The more negative the FIM (curvature), the more sharply defined is the location of the maximum
  - $\Rightarrow$ the fewer samples we need to obtain a good estimate of $\theta^*$

Review of Central Limit Theorem

If $z_1, \ldots, z_n$ are i.i.d. RVs with mean $E[z_i] = 0$ and variance $E[z_i^2] = \sigma^2$, then $\frac{1}{\sqrt{n}} \sum_{i=1}^{n} z_i \sim_{asypmp} N(0, \sigma^2)$, meaning the summation of them has a distribution that tends to a Gaussian.

Lecture 9. Generalized Linear Models

Linear Modeling Approach

Consider a labeled dataset $\{x_i, y_i\}_{i=1}^{n}$. Suppose $y_i|x_i \sim N(f_w(x_i), 1)$, meaning that $E[y_i|x_i] = f_w(x_i)$, where $f_w$ is a function parameterized by $w$.

- Log-likelihood of $w$ is $C(w) = -\frac{1}{2} \sum_{i=1}^{n} (y_i - f_w(x_i))^2 + C$
  - Thus, the MLE of $w$ is given by the least squares optimization: $\hat{w} = \arg \min_{w} \sum_{i=1}^{n} (y_i - f_w(x_i))^2$
- If we assume a linear model, i.e., $f_w(x_i) = w^T x_i$, then we have the classical least squares problem
  - And the MLE is a solution to the linear system $X^T X w = X^T y$

Generalized linear models (GLM) extend this linear modeling approach by allowing the conditional probability density to take the exponential family form $p(y|x) \propto e^{l(y,w^T x)}$, where $l(y, w^T x)$ is a convex function of $w$.

The Exponential Family Models

The exponential family is a class of distributions with the form:

$$p(y|\theta) = b(y) \cdot e^{\theta l(y) - A(\theta)}$$

- The parameter $\theta$ is called the natural parameter of the distribution
$
\begin{itemize}
  \item $t(y)$ is a sufficient static
  \item $e^{-\theta(y)}$ is a normalization constant to ensure that the probability sums/integrates to 1
    \begin{align*}
    \Rightarrow \int p(y|\theta)dy & = e^{-\mu(y)} \int b(y)e^{\theta t(y)}dy = 1 \\
    \Rightarrow a(\theta) & = \log(\int b(y)e^{\theta t(y)}dy) \\
    a(\theta) & \text{is called the log partition function}
    \end{align*}
  \item $b(y)$ is the non-negative base measure; in many cases it is equal to 1
  \item We take $\theta$ here as a parametric function of $x$, e.g., $\theta = w^T x$ as a linear model
  \item The negative log-likelihood of $\theta$ is $-\log p(y|\theta) = -\theta^T t(y) + a(\theta) - \log b(y)$
    \begin{align*}
    \text{This is a convex function of } \theta \\
    \text{The first term is linear and hence convex in } \theta \\
    \text{It can be shown that } a(\theta) \text{ is convex in } \theta
    \end{align*}
\end{itemize}

**Generalized Linear Model Examples**

Many classic distribution models can be expressed in this GLM framework:

\begin{itemize}
  \item Gaussian: $p(y|\theta) = \frac{1}{\sqrt{2\pi}} e^{-\frac{1}{2}(y-\mu)^2} = \frac{1}{2\pi} e^{-\frac{1}{2}y^2} \cdot e^{\mu y - \frac{\mu^2}{2}}$
    \begin{itemize}
    \item $b(y)$ is the first part, $\theta = \mu$, $t(y) = y$, $a(\theta) = \frac{\theta^2}{2}$
    \item This maps to classical least squares problem if we let $\theta = w^T x$
    \end{itemize}
  \item Binomial: $p(y|\mu) = \mu^y(1-\mu)^{1-y} = e^{y \log \mu + (1-y) \log(1-\mu)} = e^{y \log(\frac{\mu}{1-\mu}) + \log(1-\mu)}$
    \begin{itemize}
    \item $b(y) = 1, \theta = \log(\frac{\mu}{1-\mu})$, $t(y) = y$, $a(\theta) = \log(1 + e^\theta)$
    \item This maps to logistic regression because the function $f(\theta) = \log(\frac{1}{1+e^{-\theta}})$ is known as the logistic function
    \end{itemize}
  \item Multinomial: $p(y_1, \ldots, y_m) = \sum_{k=1}^m q_k e^{\theta^T(y_k - \xi)}$, where:
    \begin{itemize}
    \item $y_k$ is an RV that takes value $k$ with probability $p(y = k) = q_k$ for $k = 1, \ldots, m$
    \item $b(y) = 1$, $t(y) = 1$, $a(\theta) = \log(\sum_{k=1}^m e^{\theta_k})$
    \item This maps to multinomial logistic regression problem
    \end{itemize}
  \item Exponential: $p(y|\mu) = \frac{1}{\mu} e^{-\frac{y}{\mu}} = e^{\frac{1}{\mu}(-y) + \ln \frac{1}{\mu}}$
    \begin{itemize}
    \item $b(y) = 1, \theta = \frac{1}{\mu}$, $t(y) = -y$, $a(\theta) = -\ln \theta$
    \item $E[y] = \int_{-\infty}^\infty \frac{1}{\mu} e^{-\frac{t}{\mu}} dt = \mu$
    \end{itemize}
\end{itemize}

**Lecture 10. Linear Models Optimization**

**Common Loss Functions**

In GLM, assume $\theta = w^T x$, recall that we have $p(y|w^T x) \propto e^{-l(y, w^T x)}$.

\begin{itemize}
  \item The $l$ function here acts as a loss function that measures the error/distortion between $y_i$ and the value predicted by $w^T x$
    \begin{itemize}
    \item $l: \mathbb{R} \rightarrow [0, \infty)$
    \end{itemize}
  \item The general form of the optimization is finding the MLE $\hat{\theta} = \arg \min_w \sum_{i=1}^n l(y_i, w^T x_i)$
\end{itemize}

Common valid loss functions include:

\begin{itemize}
  \item **Quadratic/Gaussian**: $(y_i - w^T x_i)^2$
    \begin{itemize}
    \item In the context of binary classification, $l = (1 - y_i w^T x_i)^2$
    \end{itemize}
  \item **Absolute/Laplacian**: $|y_i - w^T x_i|$
    \begin{itemize}
    \item In the context of binary classification, $l = |1 - y_i w^T x_i|$
    \end{itemize}
  \item **Logistic**: $\log(1 + e^{-y_i w^T x_i})$
  \item **Hinge**: $\max(0, 1 - y_i w^T x_i)$
  \item **0/1-loss**: $\mathbb{1}_{\{y_i w^T x_i < 0\}}$
\end{itemize}
Comparison in binary classification context:

- This is a non-convex loss function, but is ideal in binary classification context since its expected value is exactly the probability of error.
- Other loss functions can be viewed as convex approximations to the 0/1-loss.

Optimization Approaches

In general the above optimization problem does not have a closed-form solution.

- We need to solve it by gradient descent (GD) or other iterative algorithms; say start from an initial \( w_0 \)
  - In each iteration: \( w_t = w_{t-1} - \gamma \sum_{i=1}^{n} \nabla l(y_i, w_{t-1}^T x_i) \)
  - \( \gamma > 0 \) is a step size (or learning rate)
  - If \( L \) is convex in \( w \), then GD will converge to a global minimum if \( \gamma \) is sufficiently small
  - If \( L \) is continuous but non-convex, then GD may converge to a (suboptimal) local minimum
  - If \( L \) is discrete, e.g. the 0/1-loss, then GD cannot be used to solve this optimization
- In the special case of quadratic Gaussian loss, we do have a closed-form solution \( \hat{w} = (X^T X)^{-1} X^T y \)
- In practice, the dimension \( d \) is probably large and this solution is hard to compute, therefore iterative approaches such as GD are still preferable.

Lecture 11. Gradient Descent

Gradient Descent & Strong Convexity

The Landweber iteration is given by:

\[
\hat{w}_t = w_{t-1} + \gamma X^T(y - X w_{t-1}), \gamma > 0
\]

which is equivalent to a GD method using Gaussian loss that involves all data points in each iteration.

- The step size plays an important role.
  - Too big \( \Rightarrow \) may diverge; Too small \( \Rightarrow \) may take a long time.
  - It can be shown that \( ||w_t - \hat{w}|| \leq \alpha^t ||w_t - \hat{w}_0|| = O(\alpha^t) \)
    - where \( \alpha < 1 \) is the largest eigenvalue of \( (I - \gamma X^T X) \)
    - meaning the sufficient condition for convergence is \( \gamma < \frac{2}{\lambda_{\text{max}}(X^T X)} \)
    - So the error converges exponentially in \( t \).

For a measure of the "sharpness" of convexity, we have the \( \alpha \)-strongly convex notation:

- \( f(w) \) is convex if \( f(w_2) \geq f(w_1) + \nabla_w f(w_1)^T (w_2 - w_1), \forall w_1, w_2 \)
  - A convex (but not strictly convex) function is allowed to have a "flat" region.
- \( f(w) \) is \( \alpha \)-strongly convex if \( f(w_2) \geq f(w_1) + \nabla_w f(w_1)^T (w_2 - w_1) + \frac{\alpha}{2} ||w_1 - w_2||_2^2, \forall w_1, w_2 \)

Stochastic Gradient Descent

Incremental versions of GD process just one or a small batch of samples at each step, making them scalable to extremely large datasets and problem sizes. Stochastic gradient descent (SGD) is such an incremental version; assume taking one training example per step: \( w_t = w_{t-1} + \gamma (y_i - w_{t-1}^T x_i) x_i \)

- Choices for the training example used at each step:
Round-robin: \( i_t = \lfloor t \mod m \rfloor + 1 \)
Uniformly at random: \( i_t \sim \text{Uniform}(1, \ldots, n) \), hence the name "stochastic"

- The expected value of the gradient is equal to the full gradient in this case
- \( \mathbb{E}[\frac{\partial y_i}{\partial w_0}(w_t^i)] = \frac{1}{m} \sum_{i=1}^{n}(y_i - x_i^Tw)^2 \)
- We should anticipate the algorithm will require \( t >> d \) iterations to approach convergence, where \( d \) is the number of feature dimensions

**Subgradients for Non-differentiable \( f \)**

The idea of gradient can be extended to support convex yet non-differentiable functions.

- Recall that if \( f \) is differentiable at \( w \), for all \( u \) we have \( f(u) \geq f(w) + (u - w)^T \nabla f(w) \)
- If \( f \) is non-differentiable at \( w \), we can similarly write \( f(u) \geq f(w) + (u - w)^T v \)
  - where \( v \) is a subgradient: any vector that satisfies this inequality is a subgradient of \( f \) at \( w \)
  - The set of subgradients at \( w \) is called the differential set, denoted \( \partial f(w) \)
  - If \( f \) is differentiable at \( w \), there is only one subgradient, which is the gradient itself

**Lecture 12. Analysis of Stochastic Gradient Descent**

**General SGD Iteration Analysis**

Consider the more general problem of \( w^* = \arg \min_{w \in \mathbb{R}^d} \frac{1}{T} \sum_{t=1}^{T} f_t(w) \), where \( f_t \) is a convex function.

- In the aforementioned least-squares case, \( f_t(w) = \frac{1}{2}(y_i - x_i^Tw)^2 \)
- The general SGD iteration is given by: \( w_{t+1} = w_t - \gamma \nabla f_t(w_t) \)
  - If the training set is finite and the process makes passes over the entire training set (e.g., Round-Robin or randomized), some bounds on convergence can be analyzed

Useful bounds:

- With \( \gamma_t = \gamma \) (constant stepsize):
  \[
  \frac{1}{T} \sum_{t=1}^{T} (f_t(w_t) - f_t(w^*)) \leq \frac{||w_1 - w^*||_2^2}{2\gamma T} + \frac{\gamma}{2} G^2 \quad \text{for all } T
  \]
  - \( f_t \) is convex and \( ||\nabla f_t(w)||_2 \leq G \) for all \( t, w \)
  - \( w_1 \in \mathbb{R}^d \) is an arbitrary initial weight
  - With \( \gamma = \frac{1}{\sqrt{T}} \), we have \( \text{LHS} \leq \frac{||w_1 - w^*||_2^2 + G^2}{2\sqrt{T}} \) \quad \text{for all } T
  - Using a very small but constant stepsize may lead to slow initial convergence. One way around is to use a diminishing stepsize, say \( \gamma_t = \frac{1}{\sqrt{T}} \):
  - We first modify our iteration step to include a projection step that ensures \( w \) always satisfy \( ||w_t|| \leq B \), some magnitude bound: \( w_{t+1} = \frac{w_t - \gamma \nabla f_t(w_t)}{||w_t - \gamma \nabla f_t(w_t)||} \) if \( ||w_{t+1}|| > B \)
  - Then we have the following bound:
    \[
    \frac{1}{T} \sum_{t=1}^{T} (f_t(w_t) - f_t(w^*)) \leq \frac{2B^2 + G^2}{\sqrt{T}} \quad \text{for all } T
    \]

**Lecture 13. Bayesian Inference**

**Bayesian Inference Components**

Prior distribution \( \rightarrow \) Posterior distribution of model parameter \( \theta \):

- \( p(x|\theta) \) is the likelihood of \( \theta \) when viewed as a function of \( \theta \)
- \( p(\theta) \) is the prior probability distribution of \( \theta \), reflecting our initial knowledge about \( \theta \) without observing any data points
- \( p(x) \) is the marginal probability of \( x \), which can be viewed as a constant and is usually cancelled out when doing estimation
• $p(\theta|x) = \frac{p(x|\theta)p(\theta)}{p(x)}$ is the **posterior probability** distribution of $\theta$, reflecting the probability of different values of $\theta$ in light of the observed data point $x$
  
  ○ Compared to MLE, using the posterior for estimation allows us to incorporate our prior knowledge about $\theta$
  
  ○ *Bayesian inference* methods consider the full posterior distribution

**Maximum a Posteriori Estimator (MAP)**

Maximizing the posterior produces the **Maximum a Posteriori Estimator** (MAP): $\hat{\theta}_{MAP} = \arg \max_{\theta} p(\theta|x)$.

- $\log p(\theta|x) = \log p(x|\theta) + \log p(\theta) + \text{constant} = -\log p(\theta)$ can be viewed as a **regularization** term
- MAP biases the estimator towards $\theta$ values that are higher-weighted in the prior distribution
  
  ○ Often meaning that MAP has lower variance and thus smaller overall mean-squared error -- a bias-variance tradeoff

General technique of finding the MAP of $\theta$:

1. Given the likelihood and the prior, write out the posterior distribution $p(\theta|x) \propto p(x|\theta)p(\theta)$, or the log form, or the negative form
2. Confirm that $p(\theta|x)$ is convex (concave). Do derivative w.r.t. $\theta$ and solve for $p'(\theta|x) = 0$

Taking the mean of the posterior produces the **Posterior Mean Estimator** (PM): $\hat{\theta}_{PM} = \int \theta p(\theta|x)d\theta$.

**Bias-Variance Decomposition of MSE**

The mean-squared error (MSE) of any estimator $\hat{\theta}$ can be decomposed into:

$$\text{MSE}(\hat{\theta}) = \mathbb{E}[(\theta - \hat{\theta})^2] = \mathbb{E}[(\theta - \mathbb{E}[\hat{\theta}]) + \mathbb{E}[\hat{\theta}] - \hat{\theta})^2] = (\theta - \mathbb{E}[\hat{\theta}])^2 + \mathbb{E}[(\mathbb{E}[\hat{\theta}] - \hat{\theta})^2]$$

- The cross-term equals $0$
- $(\theta - \mathbb{E}[\hat{\theta}])^2$ is the **bias** term
- $\mathbb{E}[(\mathbb{E}[\hat{\theta}] - \hat{\theta})^2]$ is the **variance** term

Example: Twitter Poisson distribution with exponential prior $p(\theta) = \alpha e^{-\alpha \theta}, \alpha > 0$

- $\hat{\theta}_{MAP} = \frac{1}{n+\alpha} \sum_{i=1}^{n} x_i = \frac{n}{n+\alpha} \hat{\theta}_{MLE}$
- The MAP is a "shrunken" version of the MLE in this case (scales down towards $0$)
  
  ○ $\mathbb{E}[\hat{\theta}_{MAP}] = \frac{n}{n+\alpha} \theta = \frac{n}{n+\alpha} \mathbb{E}[\hat{\theta}_{MLE}]$
  
  ○ $\text{Var}(\hat{\theta}_{MAP}) = \frac{(n+\alpha)^2}{n+\alpha} \frac{\theta}{n} = \frac{(n+\alpha)^2}{n+\alpha} \text{Var}(\hat{\theta}_{MLE})$

**MVN in Bayesian Inference**

If both the prior and the likelihood are Multivariate Gaussian (MVN), then the posterior distribution is also an MVN and can be computed by simple linear-algebraic operations.

- Assume the following setting:
  
  ○ Likelihood $x|\theta \sim N(\theta, \Sigma)$
  
  ○ Prior $\theta \sim N(0, \Sigma_{\theta\theta})$

  - We can derive the Wiener filter:
    
    ○ $x = \theta + N(0, \Sigma)$ so the marginal distribution is $x \sim N(0, \Sigma + \Sigma_{\theta\theta})$
    
    ○ The cross-variance between $x$ and $\theta$, $\Sigma_{x\theta} = \Sigma_{\theta\theta}$
    
    ○ $\theta|x \sim N(\Sigma_{\theta\theta}(\Sigma + \Sigma_{\theta\theta})^{-1} x, \Sigma_{\theta\theta} - \Sigma_{\theta\theta}(\Sigma + \Sigma_{\theta\theta})^{-1} \Sigma_{\theta\theta})$
    
    ○ The MAP and PM are the same: $\hat{\theta} = \Sigma_{\theta\theta}(\Sigma + \Sigma_{\theta\theta})^{-1} x$

**Bayesian Linear Modeling**

Applying the Bayesian approach to GLMs $p(y|\theta) = p(y|x^T \theta)$, we get:

- Posterior $p(\theta|x,y) \propto p(y|x^T \theta)e^{-\lambda_0 \theta^T \theta}$
- The MAP of $\theta$ is $\hat{\theta}_{MAP} = \arg \min_{\theta} \sum_{i=1}^{n} l(y_i, x_i^T \theta) - \log p(\theta)$
Different forms of priors $p(w)$ lead to different regularization, e.g.:
- $p(w) \propto e^{-\frac{1}{2}|w|^2}$ leads to ridge regularization $\frac{1}{2}|w|^2$
- $p(w) \propto e^{-\lambda|w|_1}$ leads to lasso regularization $\lambda||w||_1$

**Lecture 14. Proximal Gradient Algorithms**

**Proximal Operator & Soft-Thresholding**

Consider the general optimization problem $\min_w f(w) + g(w)$.
- Both $f$ and $g$ are convex, and $f$ is also differentiable
- Special cases of $g$ include the regularization term in GLMs
- If $g$ has a computationally efficient proximal operator with state-of-the-art performance, it is easy to implement proximal gradient algorithms

The **proximal operator** for this problem is defined as $\text{prox}_{g,t}(v) = \arg \min_u \left( \frac{1}{2}||u - v||^2 + t \cdot g(u) \right)$.
- The solution is a point close to input $v$ with a relatively small $g$ value
- $t$ controls the tradeoff between staying close to $v$ vs. minimizing $g$
- Example: $g(u) = ||u||_1$, then $\text{prox}_{g,t}(v) = \arg \min_u \sum_{i=1}^d \left( \frac{1}{2} (u_i - v_i)^2 + t |u_i| \right)$
  - The optimization objective is separable in the coordinates
  - There's a closed-form solution known as the **soft-threshold** operation: $\text{sign}(v_i) \max(0, |v_i| - t)$

**Special Case of Squared Error Loss**

Consider the special case where $f$ is the **squared error loss**:

$$L(w) = ||y - Xw||^2_2 + g(w) = ||y - Xw^{(k)}||^2_2 + 2(y - Xw^{(k)})^T X(w^{(k)} - w) + ||X(w^{(k)} - w)||^2_2 + g(w) \leq C + 2(y - Xw^{(k)})^T X(w^{(k)} - w) + \frac{1}{t} ||X(w^{(k)} - w)||^2_2 + g(w)$$

- **Notations:**
  - $k$ is the gradient descent iteration
  - $0 < t < \frac{1}{||X||^2}$
- Define $v = tX^T(y - X2^{(k)}) = -tX^T(X2^{(k)} - y)$
  - We can obtain $w^{(k+1)} = \arg \min_w \left( ||v + w^{(k)} - w||^2_2 + tg(w) \right)$
- Define $z_k = v + w^{(k)} - tX^T(Xw^{(k)} - y)$ which is the gradient descent iterate
  - $w^{(k+1)} = \arg \min_w \left( ||z_k - w||^2_2 + tg(w) \right)$ is in the proximal operator form
  - This sort of iterative optimization is often referred to as a **proximal point algorithm**
  - If $g = 0$, then $w^{(k+1)} = z_k$ the ordinary GD iterate

**General Proximal Gradient Algorithm**

Now let $f$ be any convex loss function, then $w^{(k+1)} = \text{prox}_{g,t}(w^{(k)} - t \cdot \nabla f(w^{(k)}))$
- $w^{(k+1)}$ minimizes the sum of $g(u)$ and a separable quadratic approximation of $f(u)$ around $w^{(k)}$
- The separability of this approximation is the key to efficient algorithms
  - If the regularization term $g$ is also separable, e.g. $||u||_1$, then we can write the optimization as a sum of individual coordinates and solve for each scalar element separately
  - In the case $g(u) = \lambda||u||_1$, we have the **iterative soft-thresholding algorithm** (ISTA)
    - Solutions to ISTA tend to be sparse vectors

Analysis shows that $L(w^{(k)}) - L(w^*) \leq \frac{1}{2t} ||w^{(0)} - w^*||^2_2 \leq \epsilon$ after $O(\frac{1}{\epsilon})$ iterations.
Lecture 15. Analysis of Soft-Thresholding

**Lasso Regression Soft-Thresholding Estimator**

In the “Lasso” regression problem, suppose that \( y \sim \mathcal{N}(Xw, \sigma^2 I) \) and that \( w \) is sparse, then under certain assumptions on \( X \), it can be proven that the solution \( \hat{w} \) is also sparse in the same locations.

- **Simplest setting:** \( X = I, y = w + \epsilon \) where \( \epsilon \sim \mathcal{N}(0, \sigma^2 I) \), the “direct” observation model
- **Its solution is the soft-thresholding estimator** \( \hat{w}_i = \text{sign}(y_i) \max(|y_i| - \lambda, 0) \), \( \lambda > 0 \) which is much more computationally efficient if \( w \) is sparse

---

Lecture 16. Concentration Inequalities

**Central Limit Theorem**

The **Central Limit Theorem** (CLT) is a classic result showing that the probability of the average of \( n \) i.i.d. RVs \( \bar{X}_n = \frac{1}{n} \sum_{i=1}^{n} X_i \) tends to (i.e., \( \lim_{n \to \infty} \)) a Gaussian with mean \( \mu \) and variance \( \frac{\sigma^2}{n} \).

**Chebyshev’s Inequality**

In many applications, we would like to say more about the distributional characteristics for finite values of \( n \).

- **One approach is to calculate the distribution of the average explicitly (a convolution), which is sometimes difficult or impossible**
- **Sometimes probability bounds are more useful:**
  - **Markov’s Inequality:** Let \( Z \) be non-negative RV and \( t > 0 \), \( P(Z \geq t) \leq \frac{E[Z]}{t} \)
  - A generalization of Markov: Let \( \phi \) be any non-decreasing, non-negative function,
    \( P(Z \geq t) = P(\phi(Z) \geq \phi(t)) \leq \frac{E[\phi(Z)]}{\phi(t)} \)
  - This leads to **Chebyshev’s Inequality:** Let \( t > 0 \).
    \[
    P(|Z - E[Z]| \geq t) \leq \frac{E[(Z - E[Z])^2]}{t^2} = \frac{\text{Var}(Z)}{t^2}
    \]
  - Applying Chebyshev’s to the average, we have \( P(|\bar{X}_n - \mu| \geq t) \leq \frac{\sigma^2}{nt^2} \)
  - This shows that not only is the variance reduced by average, but the “tails” of the distribution (i.e., probability of observing values more than \( t \) away from the mean) are getting smaller

Chebyshev’s tail bound is loose. Under slightly stronger assumptions, much tighter bounds are possible:

- **Example:** \( X_i \sim \mathcal{N}(\mu, 1), \mu \sim \mathcal{N}(\mu, \frac{1}{n}) \), it can be proven that \( P(|\bar{X}_n - \mu| \geq t) \leq e^{-\frac{1}{2}nt^2} \)
- **See below for examples of a few more exponential bounds**

**The Chernoff Method**

More generally, if RVs \( X_i \) are bounded or sub-Gaussian (meaning the tails of probability distribution decay at least as fast as Gaussian tails), then the tails of their average converge exponentially fast in \( n \) -- the **Chernoff bounding method**.

- The key is to use the exponential function to generalize Markov’s: \( P(Z > t) = P(e^{\lambda Z} > e^{\lambda t}) \leq e^{-\lambda t}E[e^{\lambda Z}] \)
  - Choose \( \lambda > 0 \) to minimize this bound: \( P(Z > t) = e^{-\varphi^*(t)} \text{ where } \varphi^*(t) = \max_{s > 0} (st - \log E[e^{sZ}]) \)

Exponential bounds of this form can be derived explicitly for many classes of RVs:

- **Example:** sub-Gaussian RVs \( X_i \) where \( \exists \) constant \( c > 0 \) s.t. \( E[e^{s(X_i - E[X_i])}] \leq e^{\frac{1}{2}cs^2} \) for all \( s \in \mathbb{R} \)
  - \( P(|S_n - E[S_n]| \geq t) \leq 2e^{-t^2/(2nc)} \)
  - \( P(|\bar{X}_n - \mu| \geq t) \leq 2e^{-nt^2/(2c)} \)
- To verify the sub-Gaussian condition, use this theorem: if \( P(|X_i - E[X_i]| \geq t) \leq ae^{-\frac{1}{2}bt^2} \) holds for constants \( a \geq 1, b > 0 \), and all \( t > 0 \), then \( E[e^{s(X_i - E[X_i])}] \leq e^{las^2}b \)
- **Example:** **Hoeffding’s Inequality** for bounded RVs \( X_i \in [a_i, b_i] \)
  - \( P(|S_n - E[S_n]| \geq t) \leq 2e^{-2t^2/(\sum_{i=1}^{n}(b_i - a_i)^2)} \)
  - If all \( X_i \) are bounded by \( a \leq X_i \leq b \) then it implies that \( P(|\bar{X}_n - \mu| \geq t) \leq 2e^{-2nt^2/c} \) with \( c = (b - a)^2 \)
- For binary 0/1-valued RVs:
  - \( c = 1 \) in this case: if \( n \geq \frac{1}{2} \log(\frac{1}{\delta}) \), then we know \( P(|\bar{X}_n - \mu| > \epsilon) \leq \delta \)
This result is usually referred to as the Chernoff Bound

Azuma-Hoeffding Inequality

Hoeffding’s Inequality can be generalized in a few ways:

- Using Doob’s inequality, we can derive \( P(\max_{1 \leq k \leq n} |S_k - \mathbb{E}[S_k]| \geq t) \leq 2e^{-2t^2/(\sum_{i=1}^n (b_i-a_i)^2)} \)
- Consider a martingale sequence of RVs \( S_0, \ldots, S_n \) that satisfies \( \mathbb{E}[S_{k+1} | S_0, \ldots, S_k] = S_k \) for all \( k = 0, \ldots, n \)
  - Note that sums of 0-mean and independent RVs are a martingale sequence
  - **Azuma’s Inequality**: Let \( S_0, \ldots, S_n \) be a martingale sequence s.t. \( S_i - S_{i-1} \in [a_i, b_i] \) bounded for all \( i \), then for any \( t > 0 \), we have \( P(S_n - S_0 \geq t) \leq e^{-t^2/(2\sum_{i=1}^n (b_i-a_i)^2)} \)
  - Application example: making a bet each day with 50/50 chance of receiving 2b or losing that b; Let \( S_i \) denote the net gain on day \( i \) and let \( Y_i \in \{-1, +1\} \) be an indicator of outcome on day \( i \)
    - **Independent betting** strategy: always bet fixed \( b \) then \( S_n = b \sum_{i=1}^n Y_i \)
    - **Recursive betting** strategy: on day \( i \), bet \( pS_{i-1} \) for some \( p \in [0, 1] \), then \( S_i = S_{i-1} + pS_{i-1}Y_i \) is a martingale

KL-Based Tail Bounds

It is possible to derive tighter bounds by optimizing the exponent. If the RVs belong to the exponential family, then the resulting exponent turns out to be a KL-divergence.

- Example: i.i.d. Bernoulli RVs
  - We can derive \( \psi^*(p+\epsilon) = (p+\epsilon) \log(\frac{p+\epsilon}{1+\epsilon}) + (1-(p+\epsilon)) \log(\frac{1-(p+\epsilon)}{1-p}) = \text{KL}(p+\epsilon, p) \) by Markov’s
  - Yielding \( P(\frac{1}{n} \sum_{i=1}^n x_i - p \geq \epsilon) \leq e^{-n\text{KL}(p+\epsilon, p)} \)

**Lecture 17. Probably Approximately Correct (PAC) Learning**

**Probably Approximately Correct (PAC) Learning**

Let \( \mathcal{F} \) denote a collection of prediction rules, where each \( f \in \mathcal{F} \) is a predictor function that maps from features to labels.

The aim of **Probably Approximately Correct (PAC) Learning** is to use the training data to select \( \hat{f} \) from \( \mathcal{F} \) s.t. its predictions are probably almost as good as the best possible predictor in \( \mathcal{F} \).

- Best premise of PAC: training data are i.i.d. samples from an unknown distribution \( \mathcal{P}(x_i, y_i) \sim \text{i.i.d.} \ P \)
- Goal of PAC: select a predictor that minimizes the expected loss (i.e., risk), \( \min_{f \in \mathcal{F}} \mathbb{E}_{(x,y) \sim P} [l(y, f(x))] \)
- Most natural approach: choose \( \hat{f} \) that minimizes the errors made on training data, \( \min_{f \in \mathcal{F}} \sum_{i=1}^n l(y_i, f(x_i)) \)
  - This is called **empirical risk minimization** (ERM)
  - Note that ERM / \( n \) asymptotically approaches the risk

We assume the losses are bounded in the range \([0, c]\).

**Analysis of Empirical Risk Minimization (ERM)**

Denote \( R(f) = \mathbb{E}_{(x,y) \sim P} [l(y, f(x))] \) and \( \hat{R}(f) = \frac{1}{n} \sum_{i=1}^n l(y_i, f(x_i)) \).

- Markov/Chebyshev’s weak upper bound: \( P(|\hat{R}(f) - R(f)| > t) \leq \frac{\mathbb{E}[(\hat{R}(f) - R(f))^2]}{t^2} \leq \frac{c^2}{4nt^2} \)
- Improved using Chernoff’s bounding technique:
  - \( P(\hat{R}(f) - R(f) > t) = \inf_{\lambda > 0} P(e^{\lambda(\hat{R}(f) - R(f))} > e^t) \leq e^{-2nt^2/c^2} \)
  - \( P(|\hat{R}(f) - R(f)| > t) \leq 2e^{-2nt^2/c^2} \)

If \( \hat{R}(f) \approx R(f) \) for all \( f \in \mathcal{F} \), then the minimizer of \( \hat{R} \) should be “close to” the minimizer of \( R \).

- To guarantee this approximation, we need to consider \( P(\cup_{f \in \mathcal{F}} \{ |\hat{R}(f) - R(f)| > t \}) \)
  - This is called the union bound approach
- To bound this probability, we will assume here that \( \mathcal{F} \) is finite and denote \# functions by \( |\mathcal{F}| \)
  - \( P(\cup_{f \in \mathcal{F}} \{ |\hat{R}(f) - R(f)| > t \}) \leq \sum_{f \in \mathcal{F}} P(|\hat{R}(f) - R(f)| > t) \leq 2|\mathcal{F}|e^{-2nt^2/c^2} = \delta \)
  - i.e., \( \hat{R} \) is uniformly close to \( R \) over \( \mathcal{F} \) with probability at least \( 1 - \delta \)
  - i.e., \( R(\hat{f}) \leq \hat{R}(\hat{f}) + t \leq \hat{R}(f^*) + t \leq R(f^*) + 2t \) with probability at least \( 1 - \delta \)
Let $\varepsilon = 2t = \sqrt{2C \log(2|F|/\delta)}$

- We say $\hat{f}$ is $(\varepsilon, \delta)$-PAC: $R(\hat{f}) - R(f^*) \leq \varepsilon$ with probability at least $1 - \delta$
- The error decreases with $n$ and increases with $|F|
- If the number of samples $n = O(\log |F|)$, then the class is PAC-learnable

**Lecture 18. PAC Learning in Infinite Classes**

**Generalization of PAC to Infinite Classes**

Consider the binary classification scenario with a 0/1-loss, $c = 1$.

- The PAC bound for a finite class $F$ may be stated as:
  $$P(\max_{f \in F} |\hat{R}(f) - R(f)| \geq \varepsilon) \leq 2|F|e^{-2n\varepsilon^2}$$

- For any $\delta > 0$ and for every $f \in F$, with probability at least $1 - \delta$, $R(f) \leq \hat{R}(f) + \sqrt{\frac{\log(|F|/\delta)}{2n}}$

Now we generalize this sort of result to infinite model classes. The prime example of such classes are linear classifiers:

- For arbitrary weights $\omega$ and bias $b$, $|F| = \infty$
- However, observe that the classification result does not change while we move the hyperplane of the classifier boundary until it just touches on or more of the points
  - There are effectively at most $S(F, n) = 2\sum_{k=0}^{d} \binom{n-1}{k}$ unique linear classifiers for $n$ points in $\mathbb{R}^d$
  - $S(F, n)$ is called the shatter coefficient of $F$
- We can apply PAC on this finite quantity
  - But be careful that the quantity is data-dependent on the specific locations of $x_i$'s, i.e., the errors are no longer i.i.d. RVs

**Rademacher Complexity**

Let $F$ be infinite. The goal is to derive a bound of the form $P(\sup_{f \in F} |\hat{R}(f) - R(f)| \geq \varepsilon) \leq B(n, \varepsilon)$.

- This type of bounds are called uniform deviation bounds
- For the linear classifiers case described above, we can show that:
  $$B(n, \varepsilon) = 8S(F, n)e^{-n\varepsilon^2/32}$$

The Rademacher complexity is a standard approach to construct uniform deviation bounds.

- Let $l_i(f) \in [0, 1]$ be i.i.d. bounded RVs; here they are the prediction error using $f$ on the $i$-th example
- McDiarmid's Bounded Difference Inequality: Let $g : \mathbb{R}^n \to \mathbb{R}$ be a function satisfying:
  $$\sup_{i,j} |g(l_1, \ldots, l_{i-1}, l_i, l_{i+1}, \ldots, l_n) - g(l_1, \ldots, l_{i-1}, l_i', l_{i+1}, \ldots, l_n)| \leq c_i$$
  for some constant $c_i \geq 0$ for all $i$. Then, if $l_1, \ldots, l_n$ are i.i.d. RVs, we have:
  $$P(g(l_1, \ldots, l_n) - \mathbb{E}[g(l_1, \ldots, l_n)] \geq t) \leq e^{-2t^2/(\sum_i c_i^2)}$$
  - The function $g = \sup_{f \in F} (R(f) - \hat{R}(f))$ satisfies the assumption with $c_i = \frac{1}{n}$
  - $\Rightarrow \sup_{f \in F} (R(f) - \hat{R}(f)) \leq \mathbb{E}[\sup_{f \in F} (R(f) - \hat{R}(f))] + \sqrt{\frac{\log(1/\delta)}{2n}}$

- Then, to bound the expectation, introduce an independent “ghost sample” $l'$; By Jensen’s and by introducing a set of independent Rademacher RVs $\sigma = \{\sigma_1, \ldots, \sigma_n\}$ with $P(\sigma_i = \pm 1) = \frac{1}{2}$, we can derive:
  $$\mathbb{E}[\sup_{f \in F} (R(f) - \hat{R}(f))] \leq \mathbb{E}_{l', \sigma}[\sup_{f \in F} (R(f) - \hat{R}(f))]$$
  $$= \mathbb{E}_{l', \sigma}\left[\sup_{f \in F} \frac{1}{n} \sum_{i=1}^{n} \sigma_i (l'_i(f) - l_i(f))\right]$$
  $$\leq \mathbb{E}_{l', \sigma}\left[\sup_{f \in F} \frac{1}{n} \sum_{i=1}^{n} \sigma_i (l'_i(f) + \hat{R}(f)) \sup_{f \in F} \frac{1}{n} \sum_{i=1}^{n} \sigma_i l_i(f)\right]$$
  $$= 2\mathbb{E}_{l', \sigma}\left[\sup_{f \in F} \frac{1}{n} \sum_{i=1}^{n} \sigma_i l_i(f)\right]$$
The last expression is the **Rademacher complexity** \( \text{Rad}(l(\mathcal{F})) \) of the class \( \mathcal{F} \) with loss function \( l \).

If we take the expectation only over \( \{x_i\} \) while holding \( \{l_i\} \) fixed, we have the so-called **empirical Rademacher complexity** \( \hat{\text{Rad}}(l(\mathcal{F})) \).

Putting it all together, we derive that with probability at least \( 1 - \delta \):

- \[ \sup_{f \in \mathcal{F}} (R(f) - \hat{R}(f)) \leq \text{Rad}(l(\mathcal{F})) + \sqrt{\frac{\log(1/\delta)}{2n}} \]
- \[ \sup_{f \in \mathcal{F}} (R(f) - \hat{R}(f)) \leq \hat{\text{Rad}}(l(\mathcal{F})) + 3\sqrt{\frac{\log(2/\delta)}{2n}} \]

Applying it to binary classification, we can show \( \hat{\text{Rad}}(l(\mathcal{F})) = \mathbb{E}_e[\sup_{f \in \mathcal{F}} \frac{1}{n} \sum_{i=1}^{n} \sigma_i y_i f(x_i)] = \text{Rad}(\mathcal{F}) \).

**Lecture 19. Vapnik-Chervonkis Theory**

**Shatter Coefficient & VC Dimension**

Recall the set of linear classifiers \( f(x) = \text{sign}(w^T x + b) \in \mathcal{F}, |\mathcal{F}| = \infty \):

- However, for any finite training dataset of \( n \) examples, there are at most \( S(\mathcal{F}, n) = 2^{\sum_{k=0}^{d} \binom{n-1}{k}} \) possible ways that linear classifiers can label the dataset
- \( S(\mathcal{F}, n) \) is called the **shatter coefficient** of class \( \mathcal{F} \) of linear classifiers

More generally, for any binary classification problem:

- Each classifier produces a binary label sequence for \( n \) training examples
- \( \Rightarrow \) at most \( 2^n \) distinct sequences; but often, not all sequences can be generated by functions \( \in \mathcal{F} \)
- The **shatter coefficient** of class \( \mathcal{F} \) is defined as:
  \[ S(\mathcal{F}, n) = \max_{x_1, \ldots, x_n} |\{(f(x_1), \ldots, f(x_n)) \in \{-1, +1\}^n, f \in \mathcal{F}\}| \]
- \( S(\mathcal{F}, n) \leq 2^n \), but often it is much smaller; it measures the "effective size" of \( \mathcal{F} \) w.r.t. a finite training set of size \( n \)
- \( \log S(\mathcal{F}, n) \) measures the "effective dimension" of \( \mathcal{F} \)

The **Vapnik-Chervonenkis dimension** (VC dimension) of a class \( \mathcal{F}, V(\mathcal{F}) \), is defined as the largest integer \( k \) s.t. \( S(\mathcal{F}, k) = 2^k \).

- **Sauer's Lemma**: \( S(\mathcal{F}, n) \leq (n + 1)^{V(\mathcal{F})} \)
- \( V(\mathcal{F}) \) of linear classifiers class in \( \mathbb{R}^d = d + 1 \)

**The VC Inequality**

Let \( \mathcal{F} \) be a class of binary classifiers with shatter coefficient \( S(\mathcal{F}, n) \).

- For any \( \epsilon > 0 \), \( P(\sup_{f \in \mathcal{F}} |\hat{R}_n(f) - R(f)| \geq \epsilon) \leq 2S(\mathcal{F}, n)e^{-n\epsilon^2/8} \)
- Or equivalently for any \( \delta > 0 \), with probability at least \( 1 - \delta \),
  \[ \sup_{f \in \mathcal{F}} |\hat{R}_n(f) - R(f)| \leq \sqrt{8\log S(\mathcal{F}, n) + \log \frac{2}{\delta}}/n \]

Using Sauer’s bound, we can state a generalization bound, the **VC Inequality** states that:

- For any \( \delta > 0 \) and every \( f \in \mathcal{F} \), with probability at least \( 1 - \delta \)
  \[ R(f) \leq \hat{R}_n(f) + \sqrt{8V(\mathcal{F}) \log(n + 1) + \log \frac{1}{\delta}}/n \]

**Proof of this inequality uses Massart’s Inequality**: let \( A \subset \mathbb{R}^n \) with \( |A| < \infty \) and \( r = \max_{u \in A} |u|_2 \), then
\[ \mathbb{E}_e[\frac{1}{n} \sup_{u \in A} \sum_{i=1}^{n} \sigma_i y_i u_i] \leq \frac{r \sqrt{2 \log |A|}}{n} \]

**Lecture 20. Learning with Continuous Loss Functions**

**Generalization Bounds for Continuous Loss**

Rademacher complexity bounds are interesting only if \( \text{Rad}_n(l(\mathcal{F})) \) decays as \( n \) grows.

- For continuous loss functions, e.g.
• Hinge: \( l(y, f(x)) = \max(0, 1 - yf(x)) \)
• Logistic: \( l(y, f(x)) = \log(1 + e^{-yf(x)}) \)
• Let \( z = yf(x) \)
  • We will bound \( \text{Rad}_n(l(F)) \) in terms of \( \text{Rad}_n(F) \), and then bound \( \text{Rad}_n(F) \)
    • Assume the loss \( l \) is \( L \)-Lipschitz: \( |l(z) - l(z')| \leq L|z - z'| \), then \( \text{Rad}_n(l(F)) \leq 2L\text{Rad}_n(F) \) for the continuous convex \( l \)'s we listed above
  • Hinge and logistic losses are \( 1 \)-Lipschitz functions

Applying this to linear classifiers \( f(x) = w^T x \), with \( ||w||_2 \leq 1 \) and \( ||x||_2 \leq 1 \).

• Assumptions:
  • Let \( B^d \) be the set of such \( x \)'s, \( ||x||_2 \leq 1 \)
  • Let \( F \) be a class of linear classifiers from \( B^d \to \mathbb{R} \), \( ||w||_2 \leq 1 \)
  • Assume the loss \( l \) is \( L \)-Lipschitz
  • Then we have the bound \( \text{Rad}_n(l(F)) \leq 2L\text{Rad}_n(F) \leq \frac{2L}{\sqrt{n}} \)
  • Proof of \( \text{Rad}_n(F) \leq \frac{1}{\sqrt{n}} \) uses Cauchy-Schwarz inequality and Jensen's inequality

To conclude, we have shown:

• Assume \( y_i \in [-1, 1] \) and \( ||x||_2 \leq 1 \), and let \( \hat{w} \) be a solution to the convex optimization problem
  \[
  \min_{w: ||w|| \leq 1} \sum_{i=1}^{n} (1 - y_i w^T x_i).
  \]
  • Then with probability at least \( 1 - \delta \), \( P(y \neq \text{sign}(w^T x)) \leq \frac{1}{n} \sum_{i=1}^{n} (1 - y_i \hat{w}^T x_i) + \frac{2}{\sqrt{n}} + \sqrt{\frac{2 \log 1/\delta}{n}} \)
  • Similar arguments hold for the logistic loss -- just replace the term in the sum

Lecture 21. Introduction to Function Spaces

Function Spaces & Norm

A function space is a set of functions on \( \mathbb{R}^d \) with certain parameters/construction restrictions.

• The function space of all homogeneous linear functions is \( F = \{ f : f(x) = w^T x, ||w|| \in \mathbb{R}^d \} \)
  • We can limit this further by \( F_B = \{ f : f(x) = w^T x, ||w|| \leq B \} \)
  • \( \min_{w: ||w|| \leq B} \sum_{i=1}^{n} l(y_i, w^T x_i) = \min_{w: ||w|| \leq B} \sum_{i=1}^{n} l(y_i, w^T x_i) + \lambda_B ||w||^2 \) with an appropriate regularization parameter \( \lambda_B \)

More generally, let \( ||f|| \) denote the norm of function \( f \).

• Norms map functions to real numbers, and that
  • \( ||f|| \geq 0 \)
  • \( ||f + g|| \leq ||f|| + ||g|| \)
  • If \( ||f|| = 0 \), then \( f = 0 \)

• Norms based on integrals or derivative are common
  • \( ||f|| := \int_{-\infty}^{\infty} \sqrt{\int |f(x)|^2 dx} \)
  • Given a norm, we can define a function space \( F = \{ f : ||f|| < \infty \} \) and classes \( F_B = \{ f : ||f|| \leq B \} \)
  • Consider learning with this class, \( \min_{f: ||f|| \leq \lambda_B} \sum_{i=1}^{n} l(y_i, f(x_i)) \) or \( \min_{f: ||f|| \leq \lambda_B} \sum_{i=1}^{n} l(y_i, f(x_i)) + \lambda_B ||f||^2 \)

Constructions of Function Classes

There are many ways of constructing function spaces and classes:

• Parametric classes: the simplest way to construct a function class is in terms of a set of parameters or weights:
  • Example: a neural network layer space
    \[
    F = \{ f : f(x) = \sum_{k=1}^{K} v_k \phi(w_k^T x + b_k), w_k, v_k, b_k \in \mathbb{R} \}
    \]
  • Input weights \( w_k \), output weights \( v_k \), and biases \( b_k \) are learnable parameters
  • We could further limit this class by placing constraints on the size of weights and biases
• **Atomic classes**: combinations of atom functions
  - Consider a family of parameterized functions \( \{ \phi_w \} \) — we call these functions atoms
  - We then take weighted combinations of atoms to synthesize more complex functions
  - Examples of atoms:
    - Neurons in a neural network
    - Fourier basis functions: \( \phi_w(x) = e^{iw^T x} \)
  - Examples of atomic class:
    \[
    \mathcal{F} = \{ f : f(x) = \sum_{w \in \mathcal{W}} v(w) \phi_w(x), v(w) \in \mathbb{R}, \sum_{w \in \mathcal{W}} |v(w)|^2 \leq B \}
    \]
    \[
    \mathcal{F} = \{ f : f(x) = \int v(w) \phi_w(x) dw, \int |v(w)|^2 dw \leq B \}
    \]
• **Nonparametric classes**: given a function norm \( ||f|| \) we can define \( \mathcal{F}_B = \{ f : ||f|| \leq B \} \)
  - Examples of norms:
    - \( ||f||_C^0 = \sup_{x \in [0,1]} |f(x)| \), giving \( \mathcal{F}_B^0 \)
    - \( ||f||_C^k = \sum_{j=1}^{k} \sup_{x \in [0,1]} |f^{(j)}(x)| \), giving \( \mathcal{F}_B^k \supset \mathcal{F}_B^0 \)
  - A common approach in practice is to approximate functions in such classes with parametric or atomic models
  - The **Weierstrass theorem** states that if \( f \) is continuous on \( [0, 1] \), then for any continuous \( f : [0, 1] \to \mathbb{R} \) and any \( \epsilon > 0 \), there exists a polynomial p.s.t. \( \sup_{x \in [0,1]} |p(x) - f(x)| < \epsilon \)

**Lecture 22. Banach and Hilbert Spaces**

**Review of Vector Spaces**

A vector space \( \mathcal{F} \) is a set of elements (vectors) with **addition** and **scalar multiplication** operators satisfying: for any \( u, v, w \in \mathcal{F} \) and any scalars \( a, b \in \mathbb{R} \):

- If \( u, v \in \mathcal{F} \), then \( u + v \in \mathcal{F} \)
- \( u + v = v + u \)
- \( u + (v + w) = (u + v) + w \)
- There exists a null vector \( 0 \in \mathcal{F} \) s.t. \( v + 0 = v \), i.e., the additive identity
- There exists a scalar \( -v \in \mathcal{F} \) s.t. \( v + (-v) = 0 \)
- If \( u \in \mathcal{F} \), then \( au \in \mathcal{F} \)
- \( a(0v) = (ab)v \)
- \( 1v = v \) where 1 denotes the multiplicative identity
- \( a(u + v) = au + av \)
- \( (a - b)v = av + bv \)

Many other properties can be derived from above axioms, e.g., \( 0v = 0 \)

Examples of vector spaces:

- \( \mathbb{R} \) with \( v \in \mathbb{R}^d \) and \( v = [v_1, \ldots, v_d]^T \)
- \( C([0,1]) \) with \( v \) being any **real-valued continuous function** defined on \( [0, 1] \)
- \( C^k([0,1]) \) with \( v \) being any **real-valued continuous** and \( k \)-times **differentiable** function defined on \( [0, 1] \)
- \( P_d([0,1]) \) with \( v \) being any **polynomial** of degree \( d \) or smaller defined on \( [0, 1] \)

A non-empty subset \( S \subseteq \mathcal{F} \) is a **subspace** of \( \mathcal{F} \) if \( au = bv \in S \) for all \( u, v \in S \) and scalars \( a, b \).

- \( 0 \) is always \( \in S \)

Examples of subspaces:

- \( \{ v : v = [v_1, \ldots, v_k, 0, \ldots, 0] \in \mathbb{R}^d \} \) is a subspace of \( \mathbb{R}^d \)
- \( P_0([0,1]) \) is a subspace of \( C([0,1]) \)
- If \( S \) and \( T \) are both subspaces of \( \mathcal{F} \), then \( S \cap T \) and \( S + T = \{ v : v = u + w, u \in S, w \in T \} \) are also subspaces
- An affine subspace \( S_w \) w.r.t. a fixed vector \( w \in \mathcal{F} \) is \( \{ v : v = u + w, u \in S \} \)
A set of vectors \( \{ v_j \} \) is linearly independent (i.e., no vector in the set can be written as linear combination of the others) iff. \( \sum_j \alpha_j v_j = 0 \Rightarrow \alpha_j = 0, \forall j \).

- A set of linearly independent vectors \( \{ u_i \} \) in \( \mathcal{F} \) is a basis for subspace \( S \subseteq \mathcal{F} \) if every \( v \in S \) can be written as \( v = \sum \alpha_i u_i \).
- If \( \{ u_i \} \) is finite then the dimension of \( S \) is finite; otherwise, \( S \) is infinite-dimensional.

Examples of bases:
- For \( \mathbb{R}^d \), the set of unit vectors \( \{ e_i \}_{i=1}^d \) where \( e_i \) has 1 on the \( i \)-th entry and 0 elsewhere.
- \( P_d([0,1]) \) is \( (d+1) \)-dimensional with basis \( \{ u_i(x) \}_{i=0}^d \) where \( u_i(x) = x^i \).

### Normed Vector Spaces & Banach Spaces

A normed vector space is one equipped with a functional mapping \( \| \cdot \| : \mathcal{F} \to \mathbb{R} \) s.t. for any \( u, v \in \mathcal{F} \) and scalar \( \alpha \in \mathbb{R} \):

- \( \| u \| \geq 0 \)
- \( \| u \| = 0 \iff u = 0 \)
- \( \| \alpha u \| = |\alpha| \cdot \| u \| \)
- \( \| u + v \| \leq \| u \| + \| v \| \)

Examples of normed vector spaces:

- \( \mathbb{R}^d \) with p-norm \( \| x \|_p = \left( \sum_{i=1}^d |x_i|^p \right)^{\frac{1}{p}} \), \( p \geq 1 \)
- \( C([0,1]) \) with norm \( \| f \|_{L^\infty} = \sup_{x \in [0,1]} |f(x)| \) or \( \| f \|_{L^1} = \int_0^1 |f(x)| \, dx \) or \( \| f \|_{L^2} = \left( \int_0^1 f^2(x) \, dx \right)^{\frac{1}{2}} \)
- \( C^1([0,1]) \) with norm \( \| f \| = \sup_{x \in [0,1]} |f(x)| + \sup_{x \in [0,1]} |f'(x)| \)
- \( BV([0,1]) \) with norm \( \| f \| = \| f(0) \| + TV(f) \) where:
  - \( TV(f) = \sup_{P \subseteq \mathbb{P}} \sum_{i=0}^{n-1} \| f(x_{i+1}) - f(x_i) \| \)
  - \( \mathbb{P} \) is the set of all partitions of \( [0,1] \) and \( 0 \leq x_0 \leq \cdots \leq x_n = 1 \) are the boundaries of partition \( P \)

Given a norm, one can define \( d(u, v) = \| u - v \| \) to measure the distance between two vectors.

- A sequence \( \{ v_n \}_{n \geq 1} \) in \( \mathcal{F} \) is said to converge to \( v \in \mathcal{F} \) if \( \lim_{n \to \infty} \| v_n - v \| = 0 \)
- A subspace \( S \subseteq \mathcal{F} \) is closed iff. every convergent sequence in \( S \) has its limit point in \( S \)
- A sequence \( \{ v_n \}_{n \geq 1} \) in \( \mathcal{F} \) is Cauchy if for any \( \epsilon > 0 \), there exists \( N(\epsilon) \in \mathbb{N} \) s.t. for any \( m, n \geq N(\epsilon) \), we have \( \| v_m - v_n \| < \epsilon \)

A Banach Space is a normed vector space that is complete: every Cauchy sequence in \( \mathcal{F} \) converges to limit points in \( \mathcal{F} \).

Examples of Banach/non-Banach spaces:

- \( \mathbb{R} \) with absolute-value norm is Banach
- \( \mathbb{R}^d \) with p-norm, \( p \geq 1 \) is Banach
- \( C([0,1]) \) with norm \( \| f \|_{L^\infty} \) is Banach
- \( C^1([0,1]) \) with norm \( \| f \|_{L^1} \) is NOT Banach

### Hilbert Spaces

We can equip a vector space with an inner product operator \( \langle \cdot, \cdot \rangle \) from \( \mathcal{F} \times \mathcal{F} \to \mathbb{R} \) s.t. for any \( u, v, w \in \mathcal{F} \) and any scalar \( \alpha, \beta \):

- \( \langle u, v \rangle = \langle v, u \rangle \) (symmetry)
- \( \langle \alpha u + \beta v, w \rangle = \alpha \langle u, w \rangle + \beta \langle v, w \rangle \) (linearity)
- \( \langle v, v \rangle > 0 \) if \( v \neq 0 \) (positive-definite)

The inner product induces an intuitive norm \( \| v \| = \sqrt{\langle v, v \rangle} \). A Hilbert space is a Banach space that is complete w.r.t. this norm. Examples of Hilbert/non-Hilbert spaces:

- \( \mathbb{R}^n \) with inner product \( \langle u, v \rangle = \sum u_i v_i \) is Hilbert
- \( L^1([0,1]) \) is NOT Hilbert
- \( L^2([0,1]) \) with inner product \( \langle f, g \rangle = \int f(x) g(x) \, dx \) is Hilbert
- \( P([0,1]) \) with inner product \( \langle f, g \rangle = \int f(x) g(x) \, dx \)
  - \( P([0,1]) \) is a subspace of \( L^2([0,1]) \)
Hilbert spaces have many interesting properties related to geometric intuitions:

- **Orthogonality**: Two vectors \( u, v \in \mathcal{H} \) are orthogonal if \( \langle u, v \rangle = 0 \), denoted \( u \perp v \)
  - \( u \) is orthogonal to a subspace \( S \subseteq \mathcal{H} \) if \( u \perp v \) for all \( v \in S \)
- **Pythagorean Theorem**: If \( u \perp v \), then \( ||u + v||^2 = ||u||^2 + ||v||^2 \)
- **Parallelogram Law**: For any \( u, v \in \mathcal{H} \), \( ||u + v||^2 + ||u - v||^2 = 2(||u||^2 + ||v||^2) \)

### Lecture 23. Reproducing Kernel Hilbert Spaces

**Reproducing Kernel Hilbert Space (RKHS)**

A Hilbert space \( \mathcal{H} \) of functions on domain \( X \) is said to be a **Reproducing Kernel Hilbert Space (RKHS)** if there is a function \( k \) defined on \( X \times X \)’s.t.:

- \( k(\cdot, x) \in \mathcal{H} \) for all \( x \in X \)
- \( \langle f, k(\cdot, x) \rangle = f(x) \) for all \( f \in \mathcal{H} \)
  - \( \langle k(\cdot, x'), k(\cdot, x) \rangle = k(x, x') \)
- Such a function \( k \) is called a **reproducing kernel**

Examples of RKHS and their kernel:

- \( \mathbb{R}^d \): domain \( X = \{1, \ldots, d\} \)
  - \( k(i, j) = 1 \) if \( i = j \) and 0 otherwise
- \( \mathcal{H}^1[0, 1] = \{ f : [0, 1] \to \mathbb{R}, f(0) = 0, \|f\|_X < \infty \} \) with inner product \( \langle f, g \rangle = \int f^{(1)}(u)g^{(1)}(u)du \)
  - \( k(x, x') = \min(x, x') = \int_0^x 1_{[0,x']}(u)du \)

**Construction of RKHS**

We can construct an RKHS by starting with a **positive-semidefinite (PSD) kernel function** \( k : X \times X \to \mathbb{R} \) where for all \( n \geq 1 \) and \( \{x_i\}_{i=1}^n \subseteq X \), the \( n \times n \) matrix \( K_{ij} = k(x_i, x_j) \) is PSD.

- Consider functions of the form \( f(\cdot) = \sum_{i=1}^n \alpha_i k(\cdot, x_i) \)
  - The set of all such functions is a vector space, denoted as \( \tilde{\mathcal{H}} \)
- Define the inner product on \( \tilde{\mathcal{H}} \) as \( \langle f, \tilde{f} \rangle_{\tilde{\mathcal{H}}} := \sum_{i=1}^n \sum_{j=1}^n \alpha_i \tilde{\alpha}_j k(x_i, \tilde{x}_j) \)
- We complete \( \tilde{\mathcal{H}} \) by including limits of all Cauchy sequences in \( \tilde{\mathcal{H}} \) and thus get \( \mathcal{H} \) which is an RKHS
  - The inner-product norm is \( \|f\|_H = \sum_{i,j=1}^n \alpha_i \alpha_j k(x_i, x_j) \)
- Any RKHS has a unique kernel \( k \)

Examples of PSD kernels:

- **Linear kernel**: \( X = \mathbb{R}^d \)
  - \( k(x_1, x_2) = \langle x_1, x_2 \rangle = x_1^T x_2 \)
  - \( f(x) = \sum_{i=1}^n \alpha_i k(x, x_i) = (\sum_{i=1}^n \alpha_i x_i^T) x \)
- **Polynomial kernel**: \( X = \mathbb{R}^d \)
  - \( k(x_1, x_2) = (\langle x_1, x_2 \rangle)^p = (x_1^T x_2)^p \)
    - Consider the case of \( p = 2 \), \( k(x_1, x_2) = \langle \phi(x_1), \phi(x_2) \rangle \) where \( \phi(x) = \begin{bmatrix} x_1^2 \\ x_2^2 \\ \sqrt{2} x_1 x_2 \end{bmatrix} \) is a **feature map**
    - Here \( p = 2 \), meaning a 3-dimensional feature \( \phi(x) = \begin{bmatrix} x_1^2 \\ x_2^2 \\ \sqrt{2} x_1 x_2 \end{bmatrix} \)
      - By mapping data to higher-dimensional features, previously non-linearly-separable data may become linearly-separable
  - \( k(x_1, x_2) = P(\langle x_1, x_2 \rangle) \), i.e., a polynomial of \( x_1^T x_2 \); Example: \( (1 + x_1^T x_2)^7 \)
These map to higher-dimensional features, e.g., \( \phi(x) = \frac{1}{\sqrt{2x_d}} \)

\[
    f(x) = \sum_{i=1}^{n} \alpha_i k(x, x_i) = (\sum_{i=1}^{n} \alpha_i \phi(x_i))^T \phi(x)
\]

- **Gaussian kernel:** let \( \alpha > 0 \)
  - \( k(x_1, x_2) = e^{-\alpha \|x_1 - x_2\|^2} \)
- **Laplace kernel:**
  - \( k(x_1, x_2) = e^{-\alpha \|x_1 - x_2\|} \)

**The Representer Theorem**

Let us consider the problem of learning in a potentially infinite RKHS \( \mathcal{H} \) with kernel \( k \), where the goal is to find a function \( f \in \mathcal{H} \) that best fits the set of training data and has a small norm.

For any data \( \{ (x_i, y_i) \}_{i=1}^{n} \) and any continuous loss function \( l \), the **representer theorem** states that:

- There exists \( f \in \mathcal{H} \) that minimizes \( \sum_{i=1}^{n} l(y_i, f(x_i)) + \lambda \|f\|_{\mathcal{H}}^2 \), \( \lambda > 0 \)
- And that \( f \) has a representation \( f(\cdot) = \sum_{i=1}^{n} \alpha_i k(\cdot, x_i) \), where \( \alpha_1, \ldots, \alpha_n \in \mathbb{R} \)
  - In other words, the solution is a linear combination of the functions \( k(\cdot, x_1), \ldots, k(\cdot, x_n) \)
  - All our previous results in finite-parameters linear modeling can apply in the RKHS setting -- this is referred to as the **kernel trick**
- If the loss function \( l \) is convex, the solution is unique

Let \( K \) denote the \( n \times n \) matrix with \( i, j \)-th entry \( k(x_i, x_j) \) and let \( \alpha \in \mathbb{R}^n \) be a vector with \( i \)-th entry \( \alpha_i \). We can then write the norm as \( \|f\|_{\mathcal{H}} = \alpha^T K \alpha \). We can find the solution by solving the optimization problem:

\[
    \min_{\alpha \in \mathbb{R}^n} \sum_{i=1}^{n} l(y_i, \sum_{j=1}^{n} \alpha_j k(x_i, x_j)) + \alpha^T K \alpha
\]

using techniques such as gradient descent.

**Lecture 24. Analysis of RKHS Methods**

**Rademacher Complexity Bounds**

The representer theorem shows that \( \hat{f}(\cdot) = \sum_{i=1}^{n} \hat{\alpha}_i k(\cdot, x_i) \) is a solution to \( \min_{f \in \mathcal{H}} \sum_{i=1}^{n} l(y_i, f(x_i)) + \lambda \|f\|_{\mathcal{H}}^2 \).

Recall that let loss \( l \) be an \( L \)-Lipschitz function.

- The Rademacher complexity gives that with probability at least \( 1 - \delta \):
  \[
  \sup_{f \in \mathcal{F}} (R(f) - \hat{R}_n(f)) \leq 2LR_{\alpha}(\mathcal{F}) + C \sqrt{\frac{\log(1/\delta)}{2n}}
  \]
  - where \( R_{\alpha}(\mathcal{F}) = \mathbb{E}[\sup_{f \in \mathcal{F}} \frac{1}{n} \sum_{i=1}^{n} \sigma_i f(x_i)] \).

Applying this to the constrained class of functions \( \mathcal{H}_B = \{ f \in \mathcal{H} : \|f\|_{\mathcal{H}} \leq B \} \):

- It yields a **generalization bound** of the following form:
  \[
  R(\hat{f}) \leq \hat{R}(\hat{f}) + 2LR_{\alpha}(\mathcal{H}_B) + C \sqrt{\frac{\log(1/\delta)}{2n}}
  \]
  - \( \hat{f} \) is the training loss minimizer function
  - \( \hat{R}(\hat{f}) \) is the train error; \( R(\hat{f}) \) is the test error

Recall that this bound requires losses be bounded in \( [0, C] \); to check this:

- The reproducing probability yields \( \|k(\cdot, x_i)\|_{\mathcal{H}}^2 \leq \sup_x k(x, x) \)
- By Cauchy-Schwartz, we have \( |y_i f(x_i)| \leq \|f\|_{\mathcal{H}} \|k(\cdot, x_i)\|_{\mathcal{H}} \leq B \sup_x \sqrt{k(x, x)} \)
- Let \( C \) be the upper bound on the loss function over the range \( [\pm \sup_x \sqrt{k(x, x)}] \), then we can bound the Rademacher complexity of \( \mathcal{H}_B \) as follows:
Put together, we have shown that with probability at least 1 − δ:

\[ \mathbb{R}(\hat{f}) \leq \hat{R}(\hat{f}) + \frac{2LB}{\sqrt{n}} \sup_x \sqrt{k(x, x)} + C \sqrt{\frac{\log(1/\delta)}{2n}} \]

- For example, on logistic or hinge loss and a radial kernel like the Gaussian or Laplacian kernel, we have
  \[ \mathbb{R}(\hat{f}) \leq \hat{R}(\hat{f}) + \frac{2B}{\sqrt{n}} + (1 + B) \sqrt{\frac{\log(1/\delta)}{2n}} \]
- In general, this analysis shows that learning is well-posed (won’t suffer from overfitting) if \( \frac{B}{\sqrt{n}} \) is small

**Fourier Transform Study on Kernel Functions**

The Rademacher complexity bound depends on the maximum value of the kernel function, but otherwise does not reflect particular characteristics of the kernel function.

- Consider translation-invariant kernels as \( k(x, x') = k(x - x') \), i.e., those that only depend on the difference between \( x \) and \( x' \)
- Using Fourier transforms, we can show that different kernels can have dramatically different decay characteristics

**Lecture 25. Neural Networks (NNs)**

**Neural Network Function Spaces**

Assume with the common activation function – Rectified Linear Unit (ReLU), defined by \( \sigma(\cdot) = \max\{0, \cdot\} \), a two-layer neural network is a function of the form:

\[ f(x) = \sum_{j=1}^{m} v_j \sigma(w_j^T x + b_j), \forall x \in \mathbb{R}^d \]

- \( v_j, w_j, b_j \) are trainable parameters
- for notational convenience we append the bias \( b_j \) to the weight vector \( w_j \) and append a 1 to \( x \) in following discussion

The set of neural network functions form a vector space:

\[ \mathcal{F} = \{ f : f(x) = \sum_{j=1}^{m} v_j \sigma(w_j^T x), m \geq 1, w_j \in \mathbb{R}^{d-1}, v_j \in \mathbb{R} \} \]

- The most common regularization norm is “weight decay”, equivalent to having 2-norm \( ||f|| = ||w||_2 \) on a vector containing all the weights of \( f \), but this is not a valid function norm
- We can scale the input and output weights of the \( j \)-th neuron by \( \alpha_j > 0 \) and \( \frac{1}{\alpha_j} \) without affecting the neural network function, giving us the optimization \( \min_{f, \alpha} \sum_{i=1}^{n} l(y_i, f(x_i)) + \frac{1}{2} \sum_{j=1}^{m} (\alpha_j^2 ||w_j||_2^2 + \alpha_j^2 ||v_j||^2) \)
  - The regularization term is smallest for \( \alpha_j^2 = ||v_j||_2/||w_j||_2 \)
  - So the solution to the optimization \( \min_{f, \alpha} \sum_{i=1}^{n} l(y_i, f(x_i)) + \lambda \sum_{j=1}^{m} ||v_j||_2^2 \) are equivalent to the above
  - \( \Rightarrow \) the “path-norm” of the network: \( ||f|| = \sum_{j=1}^{m} ||v_j||_2 \)

**ReLU Neural Network Banach Space**

Consider the 1-D case, fix \( ||w_j||_1 = 1 \) and absorb its scale into \( v_j \), the path norm is simply \( \sum_{j=1}^{m} |v_j| \):

- We can write \( f(x) = \sum_{j=1}^{m} v_j |w_j| \sigma(\frac{w_j}{|w_j|} (x + \frac{b_j}{|w_j|})) \)
- \( \Rightarrow f'(x) = \sum_{j=1}^{m} v_j |w_j| u(\frac{w_j}{|w_j|} (x + \frac{b_j}{|w_j|})) \)
- The total variation of such a function is \( TV(f') = \sum_{j=1}^{m} |v_j| |w_j| \)
  - In other words, in the 1-D case the path-norm is equal to the TV of \( f' \)
  - The Banach space of functions with derivatives of finite total variation is called \( BV^2(\mathbb{R}) \) – this is the ReLU neural network Banach space
Lecture 26. NN Approximation & Generalization Bounds

ReLU Neural Network Banach Space

Assume \(|w|_2 = 1\) and absorb its scale into \(v\), the vectors in \(\mathbb{R}^{d-1}\) satisfying \(|w|_2 = 1\) is the surface of unit sphere, denoted by \(S^d\).

- Let \(\mathcal{F}\) be the space of all functions of the form \(f(x) = \int \sigma(w^T x)dv(w)\) where \(v(w)\) is a finite measure on \(S^d\)
  - The measure \(v\) plays the role of the output weights
  - If we take the measure \(dv(w) = \sum_{j=1}^{m} v_j \delta(w - w_j)\), the integral produces the finite-width neural network \(f(x) = \sum_{j=1}^{m} v_j \sigma(w_j^T x)\)

- Split the measure into positive and negative parts \(v = v^+ + v^-\)
  - This suggests the norm \(\|f\| = \int_{S^d} dv^+(w) - \int_{S^d} dv^-(w)\)
  - For a finite-width neural network, \(\|f\| = \sum_{j=1}^{m} |v_j|\)
  - To eliminate the problem of non-uniqueness, take the infimum over this

- Equipped with this \(\|f\|\), \(\mathcal{F}\) is a Banach space written as
  \[
  \mathcal{F} = \{ f : f(x) = \int \sigma(w^T x)dv(w), \|f\| < \infty \}
  \]
  - When \(d = 1\), this is \(BV^2\) as discussed in the last section

Approximating Functions in \(\mathcal{F}\)

In general, and \(f \in \mathcal{F}\) is represented by an infinite-width neural network. In practice, we approximate it. Let \(\mathcal{F}_m\) denote the set of all neural networks with width at most \(m\).

- For any \(f \in \mathcal{F}\), consider \(\min_{f_m \in \mathcal{F}_m} \|f - f_m\|_{L^p(\Omega)}\)
  - where \(\|g\|_{L^p(\Omega)} = \int_{\Omega} |g(x)|^p dx\) for some bounded domain \(\Omega \subset \mathbb{R}^d\)
  - A small approximation error means good approximation using \(f_m\) to \(f\)

- It can be proven that there exists a constant \(C_0 > 0\) such that for every \(m \geq 1\) and any \(f \in \mathcal{F}\), there is a width-\(m\) neural network satisfying \(\|f - f_m\|_{L^p(\Omega)}^2 \leq \frac{C_0}{m}\)

Generalization Bounds for Neural Networks

Consider the class of 2-layer neural networks:

\[
\mathcal{F}_C = \{ f : f(x) = \sum_{j=1}^{m} v_j \sigma(w_j^T x), m \geq 1, \sum_{j=1}^{m} |v_j||w_j| \leq C \}
\]

- It can be shown that the empirical Rademacher complexity of \(\mathcal{F}_C\) satisfies
  \[\hat{\text{Rad}}_n(\mathcal{F}_C(x_1, \ldots, x_n)) \leq \frac{2C}{n} \sqrt{\sum_{i=1}^{n} |x_i|^2}\]

- Note that this bound does not involve \(m\) (#neurons), but rather depends on the scale of weights
  - Indicating that having a large number of neurons does not necessarily negatively impact the ability of neural networks to generalize well