Statistical Learning in Practice Notes

Autumn Mapes

Fall 2025

1 Part 0: Foundationals

Def: Cumulant-Generating Function

Natural logarithm of the moment-generating function:

$$K(t) = \log \mathbb{E}[e^{tX}]$$

Generates cumulants: κ_n. Note that:
κ_1 is the mean
κ_2 is the variance
κ_3 is the skewness
But then it gets weirder.

Card ID: 1770113248073

Def: Column Space

Interpreting some $m \times n$ matrix A as n column vectors, what do they span?

$$c_1 v_1 + ... + c_n v_n$$

In other words, all possible products Ax for any $x \in \mathbb{R}^n$.

Card ID: 1770238501206

What is the orthogonal complement of the column space?

The orthogonal complement of the column space is the left null space:

$$v \in \text{col}(X)^{\perp} \iff v^{\mathrm{T}} X = 0^{\mathrm{T}}$$

Card ID: 1770238501208

Def: PDF of the Multivariate Normal Distribution

$$(2\pi)^{-n/2}|\Sigma|^{-1/2}\exp\left(-\frac{1}{2}(x-\mu)^{\mathrm{T}}\Sigma^{-1}(x-\mu)\right)$$

where Σ is the covariance matrix between the components.

Note that when the components are independent, we get just a diagonal Σ with $\Sigma_{ii} = \sigma_i^2$, and this simplifies to just a product of univariate PDFs.

Card ID: 1770238501210

Def: $\frac{\partial}{\partial v}Xv$

$$X$$

Card ID: 1770238501212

Def: $\frac{\partial}{\partial v}v^{\mathrm{T}}X$

$$X^{\mathrm{T}}$$

Card ID: 1770238501213

Def: $\frac{\partial}{\partial v}v^{\mathrm{T}}Xv$

$$\frac{\partial}{\partial v}v^{\mathrm{T}}Xv = (X + X^{\mathrm{T}})v$$

Card ID: 1770238501214

Def: Var(Xv) = Cov(Xv)

$$\mathrm{Var}(Xv) = X\Sigma X^{\mathrm{T}}$$

where Σ is the covariance matrix of the random vector v.

Card ID: 1770238501216

Def: Covariance Properties

1. $\forall a$, $\text{Cov}(X, a) = 0$
2. $\forall a, b$, $\text{Cov}(X + a, Y + b) = \text{Cov}(X, Y)$
3. $\forall a, b$, $\text{Cov}(aX, bY) = ab\text{Cov}(X, Y)$ 4. $\text{Cov}(X + Y, Z + W) = \text{Cov}(X, Z) + \text{Cov}(X, W) + \text{Cov}(Y, Z) + \text{Cov}(Y, W)$

Card ID: 1770679535633

Def: Taylor Expansion

For f around x_0 for nearby x

$$f(x) \approx f(x_0) + f'(x_0)(x - x_0) + \frac{1}{2}f''(x_0)(x - x_0)^2 + ...$$
$$= \sum_{n=0}^{\infty} \frac{1}{n!} f^{(n)}(x_0)(x - x_0)^n$$

Card ID: 1770998663387

Def: Maclaurin Expansion of e^x

$$e^x = 1 + x + \frac{1}{2}x^2 + ...$$
$$= \sum_{n=0}^{\infty} \frac{x^n}{n!}$$

Card ID: 1770998663389

Def: Wilkes' Theorem

The likelihood ratio test (LRT) statistic

$$D = -2 \log \left(\frac{\text{Likelihood for null model}}{\text{Likelihood for alternative model}} \right)$$

is asymptotically distributed with the chi-squared distribution with $\dim \Theta - \dim \Theta_0$:

$$D \sim \chi^2_{\dim \Theta - \dim \Theta_0}$$

Card ID: 1770998663390

2 Part 1: Regression

2.1 Normal Linear Regression

Def: Normal Linear Model

$$Y_i = x_i^{\mathrm{T}}\beta + \varepsilon_i$$

with $\varepsilon_i \sim \mathrm{N}(0, \sigma^2)$ independent

Alternatively written $Y = X\beta + \varepsilon$

Note our assumptions: Y_i are independent and normally distributed; linear in X.

Card ID: 1770113248088

MLE for the slope of a normal linear model

$$\hat{\beta} = (X^{\mathrm{T}}X)^{-1}X^{\mathrm{T}}Y$$

(By the Gauss-Markov Theorem, $\hat{\beta}$ and $\hat{\sigma}^2$ is the best linear unbiased estimator!)

Card ID: 1770113248101

Def: The two MLEs for the variance σ^2 of a normal linear model

$$\tilde{\sigma}^2 = \frac{1}{n}||Y - X\hat{\beta}||_2^2$$

or if we'd rather have an unbiased σ^2,

$$\hat{\sigma}^2 = \frac{1}{n-p}||Y - X\hat{\beta}||_2^2$$

(By the Gauss-Markov Theorem, $\hat{\beta}$ and $\hat{\sigma}^2$ is the best linear unbiased estimator!)

Card ID: 1770113248114

2.2 Exponential Dispersion Families

Def: Exponential Dispersion Family

$$\{\mathbb{P}_{\theta,\phi} : \theta \in \Theta, \phi \in \Phi\}$$

$\Theta \subset \mathbb{R}$, $\Phi \subset (0, \infty)$, fulfilling that, for each $\mathbb{P}_{\theta,\phi}$, we have $(\forall y \in \mathbb{R})$:

$$f(y; \theta, \phi) = f_0(y, \phi) \exp\left\{\frac{y\theta - K(\theta)}{\phi}\right\}$$

where
$K : \Theta \to \mathbb{R}$ is called the cumulant function (it's just here to normalize), and $\{f_0(\cdot, \phi) : \phi \in \Phi\}$ is some family of density functions (that gets 'tilted' by the exponential portion).

Card ID: 1770113248125

What is the cumulant-generating function for an exponential diffusion family?

$$K(t) = \log \mathbb{E}[e^{ty}] = \frac{K(\phi t + \theta) - K(\theta)}{\phi}$$

Card ID: 1770113248149

What is the mean and variance of the exponential distribution family in terms of the cumulant function $K(\theta)$? What conclusions can we draw about θ?

$$\mathbb{E}_{\theta,\phi}[y] = K'(\theta)$$
$$\mathrm{Var}_{\theta,\phi}[y] = \phi K''(\theta)$$

(this can be computed simply by taking the first or second derivative of the cumulant-generating function and plugging in $t = 0$)

If we assume our variance is nonzero, and since ϕ must be positive, it must be the case then that $K''(\theta) > 0$. Thus $K'(\theta)$ must be strictly increasing and therefore invertible. So, considering the mean equation above, it makes sense to reparameterize everything in terms of $\mu := K'(\theta)$!

Card ID: 1770113248160

State the cumulant function for the EDF of a normal random variable.

$$K(\theta) = \frac{1}{2}\theta^2$$

Note that this implies

$$\mu = \frac{d}{d\theta}K(\theta) = \theta$$

and

$$\sigma^2 = \phi\frac{d^2}{d\theta^2}K(\theta) = \phi$$

Card ID: 1775594408201

State the base function $f_0(y;\phi)$ of the normal EDF

$$f_0(y;\phi) = \frac{1}{\sqrt{2\pi\phi}}\exp\left(-\frac{y^2}{2\phi}\right)$$

Card ID: 1775594408204

State the variance function of the normal EDF

$$V(\mu) = \mu$$

Card ID: 1777460526773

What is the Logit function?

$$\text{logit}(x) = \log\left(\frac{x}{1-x}\right)$$

Card ID: 1770193655774

What is the Expit function?

$$\text{expit}(x) = \frac{1}{1+e^{-x}} = \frac{e^x}{1+e^x}$$

Card ID: 1770193655775

How can you find the θ for the EDF representing $\frac{1}{n}\text{Bin}(n,p)$

$$\theta = \text{logit}(p) = \log\frac{p}{1-p}$$

and so

$$p = \text{expit}(\theta) = \frac{e^\theta}{1+e^\theta}$$

Card ID: 1775594408207

How can you find the ϕ for the EDF representing $\frac{1}{n}\text{Bin}(n,p)$

$$\phi = \frac{1}{n}$$

and so

$$n = \frac{1}{\phi}$$

Card ID: 1775594408209

State the cumulant function of the EDF for $\frac{1}{n}\text{Bin}(n,p)$

$$K(\theta) = \log(1+e^\theta) - \log(2)$$

Card ID: 1775594408210

State the normalizing $f_0(y;\phi)$ for the EDF of $\frac{1}{n}\text{Bin}(n,p)$

$$\binom{n}{yn}2^{-n}$$

Card ID: 1775594408211

State the variance function for the binomial EDF

$$V(\mu) = \mu(1-\mu)$$

Card ID: 1777460526778

State the cumulant function for the Poisson EDF

$$K(\theta) = e^{\theta} - 1$$

Card ID: 1777460526781

How can you find the θ for the EDF representing Poisson(λ)

$$\theta = \log(\mu)$$
$$\mu = e^{\theta}$$

Card ID: 1777460526783

How can you find the ϕ for the EDF representing Poisson(λ)

It's always one:

$$\phi = 1$$

Card ID: 1777460526785

Variance function for the Poisson EDF

$$V(\mu) = e^{\log \mu} = \mu$$

Card ID: 1777460526787

So, the big idea is to allow for variance that depends on the mean μ.

2.3 Generalized Linear Models

Def: Generalized Linear Model

Independent observations (x_i, Y_i) follow a generalized linear model if, for some exponential dispersion family $\{\mathbb{P}_{\mu,\phi} : \mu \in \mathcal{M}, \phi \in \Phi\}$:

1. $Y_i \sim \mathbb{P}_{\mu_i,\phi_i}$ with $\mu_i \in \mathcal{M}$ and $\phi_i = \phi a$ with known $a_1 ... a_n > 0$ and possibly unknown $\phi > 0$,

2. $g(\mu_i) = x_i^{\mathrm{T}} \beta$ for some unknown $\beta \in \mathbb{R}^p$ with a strictly monotonic twice differentiable known 'link function' $g : \mathcal{M} \to \mathbb{R}$.
(linearity after applying the link function)

Card ID: 1770113248214

Def: Log Likelihood of a Generalized Linear Model

given $Y = (Y_1, ..., Y_n)^{\mathrm{T}}$, β.

First define $\theta_i = (K')^{-1}(g^{-1}(x_i^{\mathrm{T}} \beta))$. We have

$$\begin{aligned} \mathcal{L}(\beta, \phi) &= \sum_{i=1}^{n} \log f(Y_i; \theta_i, \phi_i) \\ &= \sum_{i=1}^{n} \log f_0(Y_i, \phi a_i) + \sum_{i=1}^{n} \frac{1}{\phi a_i} \{Y_i \theta_i - K(\theta_i)\} \end{aligned}$$

Card ID: 1770193655778

Prop

If an MLE exists for the Generalized Linear Model with the canonical link, it must be unique.

If an MLE

$$\hat{\beta} = \underset{\beta \in \mathbb{R}^p}{\operatorname{argmax}}\ \mathcal{L}(\beta, \phi)$$

exists, then it must satisfy the score equation:

$$\frac{\partial \mathcal{L}(\hat{\beta}, \phi)}{\partial \beta} = 0.$$

Note that by our assumptions, we have $\theta_i = x_i^{\mathrm{T}} \beta$, yielding:

$$\frac{\partial^2 \mathcal{L}(\beta, \phi)}{\partial \beta \partial \beta^{\mathrm{T}}} = -\sum_{i=1}^{n} \frac{x_i x_i^{\mathrm{T}}}{\phi a_i} K''(x_i^{\mathrm{T}} \beta)$$

Since $K'' > 0$ and X is assumed to have full rank, the Hessian must be negative definite, and so if it exists, the MLE must be unique.

Card ID: 1770193655780

Def: Generalized Pearson Chi-Squared Statistic for ϕ

$$\hat{\phi} = \frac{\chi^2}{n - p}$$

$$\chi^2 = \sum_{i=1}^{n} \frac{(Y_i - K'(x_i^{\mathrm{T}} \hat{\beta}))^2}{a_i V(K'(x_i^{\mathrm{T}} \hat{\beta}))}$$

This comes from $\operatorname{Var}(Y_i) = a_i \phi V(\mu_i) \implies \phi$ plugged into the weighted OLS unbiased variance MLE

$$\hat{\sigma}^2 = \frac{\sum_i w_i (Y_i - \hat{\mu})^2}{n - p}$$

This is a consistent estimator for ϕ!

Card ID: 1770193655782

Explain how to derive Newton-Raphson

Suppose we want to solve $f(x) = 0$. If we start at some point x, we can simply step downhill a bunch of times to get x' with smaller and smaller $f(x')$ via a Taylor approximation:

$$0 = f(x') \approx f(x) + f'(x)(x' - x)$$

Rearranging yields

$$\begin{aligned} 0 &= f(x) + f'(x)(x' - x) \\ -f'(x)(x' - x) &= f(x) \\ x' &= x - \frac{f(x)}{f'(x)} \end{aligned}$$

Card ID: 1770193655784

Explain how Newton-Raphson can be used to solve an MLE numerically

---

We want to find the θ for which $\frac{dl(\vec{\theta};x)}{d\vec{\theta}} = 0$. So, use Newton-Raphson with the Hessian

$$\frac{d^2 l(\vec{\theta};x)}{d\vec{\theta} d\vec{\theta}^{\mathrm{T}}}$$

except we want the maximum, not the minimum, so make sure to use the negation instead!

$$\hat{\theta}^{(n+1)} = \hat{\theta}^{(n)} - \left(\frac{d^2 l}{d\vec{\theta} d\vec{\theta}^{\mathrm{T}}} (\hat{\theta}^{(n)}) \right)^{-1} \frac{dl}{d\vec{\theta}} (\hat{\theta}^{(n)})$$

Card ID: 1776686107885

Explain briefly what Fisher scoring is, in the context of solving a general linear model numerically

---

Similar to Newton-Raphson, but replaces the Jacobian $J(\beta)$ with the relevant Fisher information:

$$I(\beta) = \mathbb{E}(J(\beta))$$

Card ID: 1770193655786

2.4 Evaluation and Comparison of GLMs

Def: The Continuous Mapping Theorem

Given a function $g : X \to Y$ that is continuous except at points of measure zero, g preserves convergence almost surely, in probability, and in distribution.

$$X_n \xrightarrow{a.s.} X \implies g(X)_n \xrightarrow{a.s.} g(X)$$

$$X_n \xrightarrow{p} X \implies g(X)_n \xrightarrow{p} g(X)$$

$$X_n \xrightarrow{d} X \implies g(X)_n \xrightarrow{d} g(X)$$

Card ID: 1776701345619

Def: Proof: Solve for the confidence interval around some parameter j of the estimated slope $\hat{\beta}$

Note that we know

$$I(\beta)^{1/2}(\hat{\beta} - \beta) \xrightarrow{d} N(0, I_p)$$

Because $\hat{\beta} \xrightarrow{p} \beta$, we can apply the continuous mapping theorem and Slutsky's Theorem to get:

$$I(\hat{\beta})^{1/2}(\hat{\beta} - \beta) \xrightarrow{d} N(0, I_p)$$

This yields the interval

$$C_\alpha := \left[\hat{\beta}_j \pm z_{\alpha/2}\sqrt{[i(\hat{\beta})^{-1}]_{jj}}\right]$$

Card ID: 1770998663402

Def: Confidence Interval for GLMs given a new datapoint

Denote the new datapoint $x^* \in \mathbb{R}^p$, we want to find a confidence interval for $\mathbb{E}(Y|X = x^*) = g^{-1}(x^{*T}\beta)$.

$$C_\alpha := g^{-1}\left[x^{*T}\hat{\beta} \pm z_{\alpha/2}\sqrt{x^{*T}i(\hat{\beta})^{-1}x^*}\right]$$

This comes from the fact that

$$\hat{\beta} \sim N(\beta,\, i(\hat{\beta})^{-1})$$

so

$$x^{\mathrm{T}}\hat{\beta} \sim N(x^{\mathrm{T}}\beta,\, x^{\mathrm{T}}i(\hat{\beta})^{-1})$$

Card ID: 1770998663404

Def: Mean parameterization $\tilde{l}(\mu, \phi)$

$$\begin{aligned}\tilde{l}(\vec{\mu}, \phi) &= \log \tilde{f}(y, \vec{\mu}, \phi) \\ &= \log f(y, \theta(\vec{\mu}), \phi) \\ &= \sum_{i=1}^{n} \log f_0(y_i, \phi) + \sum_{i=1}^{n} \frac{1}{a_i\phi}(y_i\theta(\mu_i) - K(\theta(\mu_i))\end{aligned}$$

Also known as the 'saturated' parameterization.

Card ID: 1770998663406

Def: Deviance of a GLM

$$D(\hat{\mu}) := -2\phi(\tilde{l}(\hat{\mu}, \phi) - \tilde{l}(Y, \phi))$$

where $\tilde{l}$ is the mean parameterization of the log likelihood (where we can provide a separate μ_i for each sample i). Note then we have

$$D(\hat{\mu}) = -2\phi \log \left(\frac{\sup_{\mu \in \mathbb{R}^n : \mu_i = g^{-1}(x_i^{\mathrm{T}}\beta),\, \beta \in \mathbb{R}^p} \tilde{f}(Y; \mu, \phi)}{\sup_{\mu \in \mathbb{R}^n} \tilde{f}(Y; \mu, \phi)} \right)$$

Here, the top model is the saturated model and the bottom model is the normal restricted model. So $D(\hat{\mu})$ is the likelihood ratio test statistic for H_0: GLM, H_1: saturated model.

Also note that deviance is independent of ϕ!

Card ID: 1770998663408

Def: Dispersion-scaled Pearson Residuals

$$e_i = \frac{Y_i - \hat{\mu}_i}{\mathrm{Var}(Y_i)} = \frac{Y_i - \hat{\mu}_i}{\sqrt{\hat{\phi} a_i V(\hat{\mu}_i)}}$$

Card ID: 1770998663410

Def: Deviance Residuals

$$d_i = \mathrm{sgn}(Y_i - \hat{\mu}_i)\sqrt{D_i(\hat{\mu})}$$

where $D_i(\mu)$ is the ith summand of the deviance

Card ID: 1770998663412

What is the relationship between dispersion-scaled Pearson residuals and deviance residuals?

When $Y_i \approx \hat{\mu}_i$, we have $d_i = \hat{\phi}^{1/2} e_i + O(|Y_i - \hat{\mu}_i|^{3/2})$

Card ID: 1770998663413

When we inspect GLM residuals (Pearson or Deviance), what are we looking for?

Should both be approximately normal with mean zero, variance 1.

Note this is based on small-dispersion asymptotics:

$$\phi_i^{-1/2}(y_i - \mu_i) \xrightarrow{d} N(0, V(\mu_i))$$

as $\phi_i \to 0$.

Card ID: 1776686107890

What do we need to be careful of when using dispersion-scaled residuals in R?

They aren't normalized by $\hat{\phi}^{-1/2}$, they use:

$$\frac{y_i - \hat{\mu}_i}{\sqrt{a_i V(\mu_i)}}$$

Card ID: 1776686107892

2.5 Model Selection

2.5.1 Information Criteria

Def: Akaike Information Criterion (AIC)

$$AIC(M) = 2\dim\Theta - 2\log f(z;\hat{\theta})$$

Smaller is better!

Card ID: 1776873013147

Def: Bayesian Information Criterion (BIC)

$$BIC(M) = (\log n)\dim\Theta - 2\log f(z;\hat{\theta})$$

where n is the number of data points.
Smaller is better!

Card ID: 1776873013156

When should you use AIC versus BIC?

Use AIC in predictive applications, use BIC in explanatory applications (since it selects for simpler models; extra weight on the $\dim \Theta$ term)

Card ID: 1776873013160

2.5.2 Bias-Variance Decomposition

Def: Prediction Error Err **(Risk)**

$$\text{Err} = \mathbb{E}[L(Y^*, \hat{f}(X^*))]$$

where (X^*, Y^*) is a random vector, the test data (space of all possible outcomes) and $\hat{f}$ is an arbitrary regression function fit using training data.

Card ID: 1776873013164

Def: In-Sample Error Err_{in}

$$\text{Err}_{\text{in}} = \frac{1}{n}\sum_{i=1}^{n} \mathbb{E}[L(Y_i^*, \hat{f}(x_i))]$$

(this is for a fixed design scenario, where the x_is are fixed but the outcome is not. $\hat{f}$ is fit using specific draws (x_i, y_i) but evaluated against any possible set of Y_i^* (still at fixed x_i).

Card ID: 1776873013168

Def: Training Error Err_{tr}

$$\text{Err}_{\text{tr}} = \frac{1}{n}\sum_{i=1}^{n} L(y_i, \hat{f}(x_i))$$

The evaluation of $\hat{f}$ against the training data itself.

Card ID: 1776873013172

Show that we have the relation:

$$\mathrm{Err}_{\mathrm{in}} = \mathbb{E}(\hat{\mathrm{Err}}_{\mathrm{tr}}) + \frac{2}{n}\sum_{i=1}^{n} \mathrm{Cov}(\hat{f}(x_i), Y_i)$$

for L2 loss.

Proof sketch:
Just expand the squared terms. You'll need to use that

$$\mathbb{E}[Y_i^* \hat{f}(X_i)] = \mathbb{E}[Y_i^*]\mathbb{E}[\hat{f}(X_i)] = \mathbb{E}[Y_i]\mathbb{E}[\hat{f}(X_i)]$$

Card ID: 1776873013176

Def: Bias-Variance Tradeoff (of the prediction error)

If we're trying to model

$$Y_i = f(X_i) + \varepsilon_i$$

where $f : \mathbb{R}^p \to \mathbb{R}$, $\mathbb{E}(\varepsilon_i) = 0$, $\mathrm{Var}(\varepsilon_i) = \sigma^2 > 0$, then we have the following decomposition of prediction error:

$$\mathrm{Err} = \sigma^2 + \mathbb{E}[\mathrm{Bias}^2(\hat{f}(X^*) \,|\, X^*)] + \mathbb{E}[\mathrm{Var}(\hat{f}(X^*) \,|\, X^*)]$$

Or for the in-sample error:

$$\mathrm{Err}_{\mathrm{in}} = \sigma^2 + \frac{1}{n}\sum_{i=1}^{n} \mathbb{E}[\mathrm{Bias}^2(\hat{f}(X_i^*) \,|\, X_i^*)] + \frac{1}{n}\sum_{i=1}^{n} \mathbb{E}[\mathrm{Var}(\hat{f}(X_i^*) \,|\, X_i^*)]$$

Card ID: 1776873013180

TODO PROVE THIS!
TODO linear fit of a nonlinear function example

Model Complexity and the Bias-Variance Decomposition

Complex models have low bias, but high variance.
Simple models have high bias, but low variance.
Card ID: 1780667445383

2.6 Overdispersion

State three reasons that we might observe overdispersion in our model

1. Missing critical covariates
2. You're using the wrong distribution
3. There is correlation between your datapoints

Card ID: 1777460526808

TODO example where they show that a missing covariate can create a higher-variance-than-expected Poisson model

2.6.1 Negative Binomial EDF

Intuitively, what does the negative binomial distribution model?

In a binary experiment with probability p of failure, the number of failures before r successes is reached.

NOTE: Other places use the convention that p is the probability of success, not failure. This course uses the 'probability of failure' convention.

Card ID: 1777460526810

Def: Negative Binomial Distribution PDF

If $X \sim \text{NB}(r, p)$:

$$\mathbb{P}(X = k) = \binom{k + r - 1}{k} p^k (1 - p)^r$$

Card ID: 1777460526811

Mean of the Negative Binomial PDF

If $X \sim \text{NB}(r, p)$:

$$\mathbb{E}(X) = \frac{rp}{1 - p}$$

Card ID: 1777460526813

Variance of the Negative Binomial PDF

If $X \sim \text{NB}(r, p)$:

$$\text{Var}(X) = \frac{rp}{(1-p)^2}$$

Card ID: 1777460526815

Variance Function of the Negative Binomial PDF

$$V(\mu) = \mu + \frac{\mu^2}{r}$$

Card ID: 1777460526817

What is the relationship between the Negative Binomial distribution and the Poisson distribution?

As $r \to \infty$,

$$\text{NB}\left(r, \frac{\mu}{\mu + r}\right) \xrightarrow{d} \text{Pois}(\mu)$$

Card ID: 1777460526818

TODO EDF FORM OF THE NEGATIVE BINOMIAL

What do we need to remember when using the negative binomial in R?

The default link function is the same as for the Poisson RV:

$$g(\mu) = \log \mu$$

Card ID: 1777460526820

How can we test between H_0: Poisson model and H_1: negative binomial model?

Use the modified framework:

$$2\left(l_1(\hat{\beta}, \hat{r}; Y) - l_0(\hat{\beta}; Y)\right) \xrightarrow{d} \frac{1}{2}\delta_0 + \frac{1}{2}\chi^2_1$$

as $n \to \infty$, where δ_0 is the point mass at zero.

Card ID: 1777460526822

TODO example sheet negative binomial facts?

2.6.2 Quasi-Likelihood Models

Def: Quasi-Likelihood Model

(x_i, y_i) follow a quasi-likelihood model if they are independent and satisfy:

$$\mathbb{E}[y_i] = \mu_i(\beta) = g^{-1}(x_i^T \beta)$$

$$\text{Var}[y_i] = V_i(\mu_i(\beta))$$

where we fix V_i and g, and we try and fit β.

This effectively means that we can just fix some link function and some variance function, without having to specify anything else about our model. Every GLM fits this format.

Card ID: 1777585768187

Def: Quasi-Score

$$U(\beta) = \sum_{i=1}^{n} \frac{y_i - \mu_i(\beta)}{V_i(\mu_i(\beta))} \mu_i'(\beta) x_i$$

where

$$\mu_i'(\beta) = (g^{-1})'(x_i^{\mathrm{T}}\beta)$$

Note that this fulfills the score identity $\mathbb{E}[U(\beta)] = 0$, and the Fisher information can still be calculated the standard way, $-\mathbb{E}[\frac{du}{d\mu(\beta)}]$

We can estimate β by solving the estimating equation $u(\hat{\beta}) = 0$ (called the Quasi-Likelihood Estimator)

NOTE that we don't create another x_i term when differentiating! It's already included!

Card ID: 1777585768189

Quasi-Likelihood Estimator Distribution

$$(X^{\mathrm{T}}WX)^{1/2}(\hat{\beta} - \beta) \xrightarrow{d} N(0, I_p)$$

where W is the diagonal matrix with entries

$$W_{ii} = \frac{\mu_i'(\hat{\beta})^2}{V_i(\mu_i(\hat{\beta}))}$$

where $\hat{\beta}$ is the Quasi-Likelihood Estimator.

Card ID: 1777585768191

Can't compute AIC, BIC, deviances.

2.6.3 Mixed Effects Models

What if we have correlations between observations?
Idea: split data into groups where we suspect there may be correlations among peers; model group effects as an extra parameter.
Individuals are tied together by group-level contamination to the same factors (but may respond to different degrees)

Def: Linear Mixed Effects Model

Given observations (x_{ij}, z_{ij}, Y_{ij}), with $i \in [n]$ groups and $j \in [d]$ individuals per group,

$$Y_i = X_i\beta + Z_i u_i + \varepsilon_i$$

where $X_i \in \mathbb{R}^{d\times p}$,
where $Z_i \in \mathbb{R}^{d\times q}$,
$u_i \overset{iid}{\sim} N(0, \Sigma_u)$ are random effects (Σ_u is positive semi definite in $\mathbb{R}^{q\times q}$),
$\varepsilon_i \overset{iid}{\sim} N(0, \sigma^2 I_d)$ (independent of u_i),
$\beta \in \mathbb{R}^p$ are fixed effects

Y_i represents all the observations from group i.

Card ID: 1777585768193

Def: Hierarchial Form of Linear Mixed Effects Models

$$u_i \sim N(0, \Sigma_u)$$
$$Y_i \,|\, u_i \sim N(X_i\beta + Z_i u_i,\, \sigma^2 I_d)$$

Card ID: 1778592619761

Def: Marginal Formulation of Mixed Effects Models

Just compute expectation and variance of Y_i:

$$V_i := Z_i \Sigma_u Z_i^{\mathrm{T}} + \sigma^2 I_d$$

$$Y_i \sim N(X_i\beta,\, V_i)$$

Let $Y = (Y_1^T ... Y_n^T)$, similar for X and Z, and let $V = diag(V_1, ..., V_n)$. Taking the log-likelihood of the random multivariate normal pdf yields:

$$l(\beta, \sigma^2, \Sigma_u) = -\frac{1}{2} nd \log(2\pi) - \frac{1}{2}\log(\det(V)) - \frac{1}{2}(Y - X\beta)^T V^{-1}(Y - X\beta)$$

Card ID: 1778106432622

MLE for β in a linear mixed effects model

$$\hat{\beta} = (X^{\mathrm{T}}V^{-1}X)^{-1}X^{\mathrm{T}}V^{-1}Y$$

Card ID: 1778106432624

Def: Restricted Maximum Likelihood Estimation (ReML) for Linear Mixed Effects Models

We want a matrix L such that $LX = 0$. Use $L = I - P = I - X(X^{\mathrm{T}}X)^{-1}X^{\mathrm{T}}$, then LY simplifies nicely:

$$LY = LX\beta + L[Z_i u_i] + \varepsilon \sim N_r(0, LVL^{\mathrm{T}})$$

So just estimate σ^2, Σ_u using that as the MLE.

Card ID: 1778106432626

Def: Generalized Linear Mixed Effects Models (GLMMs)

Assume the normal setup for linear mixed effects models, but let u_i have an arbitrary distribution with unknown parameter α: $u_i \overset{d}{\sim} \mathbb{P}_\alpha$. Assumes

$$g(\mathbb{E}[Y_{ij}|u_i]) = x_{ij}^{\mathrm{T}}\beta + z_{ij}^{\mathrm{T}}u_i$$

Estimate (α, β, ϕ) by MLE:

$$f(y; \alpha, \beta, \phi) = \int f(y \mid u; \beta, \phi) f(u; \alpha) du$$

Card ID: 1778106432627

TODO NOTE ON HYPOTHESIS TESTING GLMMS

2.7 The Parametric Bootstrap

Def: The Parametric Bootstrap

Suppose we know the family of distributions relevant, $\mathbb{P}_\theta$, but not the particular $\mathbb{P}_{\theta_0}$ with some property $\psi(\theta_0)$. The idea is to estimate θ_0 with an estimator $\hat{\theta}$, then simulate a bunch of new trials to understand the distribution of ψ with $\hat{\psi}$:

1. For $b = 1...B$, sample $Y^{(b)} = (Y_1^{(b)}, ..., Y_n^{(b)}) \overset{iid}{\sim} \mathbb{P}_{\hat{\theta}}$ from the estimated $\hat{\theta}$ derived from Y.

Use this to compute $\hat{\theta}^{(b)} = \hat{\theta}(Y^{(b)})$ and plug in to get $\hat{\psi}^{(b)} = \psi(\hat{\theta}^{(b)})$.

2. Approximate $\hat{\psi}$ with the empirical distribution:

$$\hat{\mathbb{P}}^{(b)} = \frac{1}{B}\sum_{i=1}^{B} \delta_{\hat{\psi}^{(b)}}$$

Card ID: 1779308511625

Def: Percentile Bootstrap Confidence Interval

In the parametric bootstrap, simulate B trials, order by their outcome $\hat{\psi}^{(b)}$, the interval $[\hat{\psi}^{0.025B}, \hat{\psi}^{0.975B}]$ is an approximate two-tailed 95 percent confidence interval.

More formally, we denote this as $[\hat{q}_{\alpha/2}, \hat{q}_{1-\alpha/2}]$

Note that you can use the likelihood ratio test statistic as ψ, like:

$$\psi(\theta) = 2\{l(\beta, \sigma^2, \Sigma_u) - l(\beta, \sigma^2, 0)\}$$

Card ID: 1779308511627

When is the percentile bootstrap confidence interval most valid?

When we assume $\hat{\psi} \approx N(\psi_0, \sigma_\psi^2)$ and $\hat{\psi}^* \approx N(\hat{\psi}, \hat{\sigma}_\psi^2)$ conditional on Y (where $\hat{\sigma}_\psi^2 \approx \sigma_\psi^2$)

Card ID: 1779308511628

Def: Basic Bootstrap Confidence Interval

and its assumptions

$$[2\hat{\psi} - \hat{q}_{1-\alpha/2}, 2\hat{\psi} - \hat{q}_{\alpha/2}]$$

Relies on the assumption that $\hat{\psi}^* - \hat{\psi}$ under $\mathbb{P}^*$ is similar to $\hat{\psi} - \psi_0$, so that we have:

$$\begin{aligned}\mathbb{P}(2\hat{\psi} - \hat{q}_{1-\alpha/2} \leq \psi_0 \leq 2\hat{\psi} - \hat{q}_{\alpha/2}) &\leq \mathbb{P}(\hat{q}_{1-\alpha/2} - \psi \leq \hat{\psi} - \psi_0 \leq \hat{q}_{\alpha/2} - \psi) \\ &\approx \mathbb{P}^*(\hat{q}_{1-\alpha/2} - \psi \leq \hat{\psi}^* - \hat{\psi} \leq \hat{q}_{\alpha/2} - \psi) \\ &= 1 - \alpha\end{aligned}$$

Card ID: 1779308511629

2.8 Regularization

TODO bias-variance decomp for ridge regression

3 Part 2: Classification

3.1 Introduction

Def: Classifier

A function $h : \mathbb{R}^p \to [K]$, typically of the form

$$h(x) = h(x; (X_i, Y_i)_{i \in [n]})$$

where p is the dimension of the classified points, K is the number of classification groups, and n is the number of data points.

Card ID: 1775594408219

Def: Misclassification Loss

$$L(y, y') = \mathbb{1}_{\{y \neq y'\}}$$

(really obvious, just 1 when you've got the wrong classification or 0 if not)

Card ID: 1775594408220

Def: Prediction Error of a Classifier

$$R(h) = \mathbb{E}[L(Y^*, h(X^*))] = \mathbb{P}[Y^* \neq h(X^*)]$$

where $(X^*, Y^*) \in (X^K, Y^K)$ are test points

We want to find classifiers h that minimize this!

Card ID: 1775594408221

Def: Conditional Class Probabilities

$$p_k(x) = \mathbb{P}(Y^* = k \mid X^* = x)$$

where k is some classification in $[K]$.

Card ID: 1775594408223

Def: Bayes Classier

The minimal prediction error over all classifiers is called the Bayes risk. It is attained by the Bayes classifier

$$h^*(x) = \underset{k \in [K]}{\operatorname{argmax}}\ p_k^*(x)$$

Card ID: 1775594408224

Def: Probability as an Expectation

For all events A,

$$\mathbb{P}(A) = \mathbb{E}[\mathbb{1}(A)]$$

Card ID: 1779308511634

Prop

Over all classifiers, the Bayes classifier has the minimal prediction error:

$$\min_{h \in H} R(h) = R(h^*) = \mathbb{E}[1 - P_{h^*(X^*)}(X^*)]$$

where H is the set of all classifiers.

Proof sketch:
Express the classification risk as an expectation.
Use the tower property to fix some $X^* = x$, and split into a sum over possible values of Y^*.
Note that $\mathbb{1}(h(x) \neq k)$ is now deterministic, so isolate an expectation on $Y^* = k$.
Note that this is equivalent to the statement of the conditional class probability $p_k(x)$.
Clearly the resulting statement is minimized when $h(x) = p_k(x)$.

Card ID: 1779308511635

Def: Plug-in Classifier

To construct a classifier $\hat{h}(x)$, define an estimator $\hat{p}_k(x)$, then use the plug-in formula

$$\hat{h}(x) = \operatorname*{argmin}_k \hat{p}_k(x)$$

Card ID: 1779308511636

Def: Linear and nonlinear classifiers

A classifier is linear if its decision boundaries are all subsets of hyperplanes in $\mathbb{R}^p$

Card ID: 1779308511637

3.2 Linear Discriminant Analysis (LDA)

Def: Log-Odds

Log of the odds as we know it: like '3 to 1 in our favor'.
For an arbitrary probability p, the logit gives its log-odds.
For classification problems, it's always given by

$$\log\left(\frac{p_k(x)}{p_j(x)}\right)$$

Card ID: 1780146566838

Def: Class Densities and Class Prior Probabilities for LDA

Class densities: $f_k : X^* \mid (Y^* = k)$
Class prior probabilities: $\pi_k = \mathbb{P}(Y^* = k)$

Card ID: 1779308511639

Conditional class probability in terms of class densities and class prior probabilities (for LDA)

$$p_k = \frac{f_k(x)\pi_k}{\sum_{j=1}^{K} f_j(x)\pi_j}$$

Card ID: 1779308511641

Def: Assumptions for Linear Discriminant Analysis

Assume each class density $f_k = X^* \mid (Y^* = k)$ is normally distributed around its mean:

$$f_k \sim N_p(\mu_k, \Sigma)$$

Need to solve for all π_i, μ_i, $i \in [p]$ and Σ.

Card ID: 1779308511642

Def: Discriminant Functions

In linear discriminant analysis, we assume each class density is normal, and so plugging into the conditional class probability, throwing out common factors, and taking a log, we get:

$$h_k^*(x) \in \underset{k}{\text{argmax}}\ \delta_k(x)$$

$$\delta_k(x) := -\frac{1}{2}(x-\mu_k)^{\mathrm{T}}\Sigma^{-1}(x-\mu_k) + \log(\pi_k)$$

Card ID: 1779308511643

Def: Decision Boundaries of LDA Bayes

The subset $\{x \in \mathbb{R}^p : p_k(x) = p_l(x)\}$, where:

$$\log\left(\frac{p_k(x)}{p_l(x)}\right) = x^T\Sigma^{-1}(\mu_k - \mu_l) - \frac{1}{2}(\mu_k + \mu_l)^{\mathrm{T}}\Sigma^{-1}(\mu_k - \mu_l) + \log\frac{\pi_k}{\pi_l}$$

$\Sigma^{-1}(\mu_k - \mu_l)$ are the normal vectors to the hyperplanes seperating the fitted classes

Card ID: 1779308511645

Def: Assumed values of each parameter $\hat{\pi}_k$, $\hat{\mu}_k$, and $\hat{\Sigma}$ in h^{LDA}

$\hat{\pi}_k = \frac{N_k}{n}$ (where N_k is the number of observations of class k)
$\hat{\mu}_k = \frac{1}{N_k}\sum_{i:Y_i=k} X_i$ (centriod of class k)
$\hat{\Sigma} = \frac{1}{n-K}\sum_{k=1}^{K}\sum_{i:Y_i=k}(X_i - \hat{\mu}_k)(X_i - \hat{\mu}_k)^{\mathrm{T}}$ (pooled sample covariance matrix)

Used in $h^{LDA}(x)$:

$$h^{LDA}(x) \in \underset{k}{\text{argmax}}\ \hat{\delta}_k(x)$$

$$\hat{\delta}_k(x) := -\frac{1}{2}(x-\hat{\mu}_k)^{\mathrm{T}}\hat{\Sigma}^{-1}(x-\hat{\mu}_k) + \log(\hat{\pi}_k)$$

This is an MLE, and is therefore consistent.

Card ID: 1779308511646

3.3 Logistic Classifier

Def: Logistic Classifier

Suppose we have independent observations $\tilde{Y}_i \in \{1, ..., K\}$ over K classes, so each $Y_i \in \{0,1\}^k$, $Y_{ij} = \mathbb{1}_{\{\hat{Y}_{i=j}\}}$. We have:

$$Y_i \,|\, (X_i = x_i) \overset{ind}{\sim} \text{Multinomial}(1; p_1, ..., p_K(x_i))$$

Note that we usually define $p_K(x) = 1$ for all x.

Card ID: 1779308511648

Def: Conditional Class Probabilities for the Logistic Classifier

$$p_k(x_i) = \frac{e^{x_i^T \beta_k}}{1 + \sum_{j=1}^{K-1} e^{x_i^T \beta_j}}$$

Note that we set $\beta_K := 0$ as a baseline.

Card ID: 1779308511650

Def: Log-Odds of the Logistic Classifier

$$\log \frac{p_k(x_i)}{p_K(x_i)} = x_i^T \beta_k$$

with $\beta_K = 0$ as the baseline.
Extending this, we have:

$$\log \left(\frac{p_k(x_i)}{p_l(x_i)} \right) = x_i^{\mathrm{T}} (\beta_k - \beta_l)$$

Card ID: 1780146566847

Def: Logistic Classifier Log Likelihood

$$L(\beta_1, ..., \beta_{K-1}) = \log \left(\prod_{i=1}^{n} \prod_{k=1}^{K} p_k(x_i)^{Y_{ik}} \right)$$

Card ID: 1779308511652

When should you use LDA versus the logisitic classifier?

LDA yields smaller variance because of the Gaussian assumption, however the logistic classifier is more robust if Gaussianity does not hold

Card ID: 1779308511654

3.4 Nearest Neighbors (KNN)

What do we mean by $X_{(i)}(x)$ and a corresponding $Y_{(i)}$?

It's just a permutation of the data, sorting by distance to a point x:

$$||X_{(1)}(x) - x|| \leq ||X_{(2)}(x) - x|| \leq ... \leq ||X_{(n)}(x) - x||$$

Card ID: 1778592619769

Def: L-Nearest Neighbors Classifier

We have the conditional class probabilities:

$$\hat{p}_k(x) = \frac{1}{L} \sum_{i=1}^{L} \mathbb{1}_{Y_{(i)}=k}$$

Then we just use the plug-in estimation for the classifier:

$$h^{LNN}(x) := \underset{k}{\text{argmax}}\ \hat{p}_k(x)$$

(majority vote among L nearest neighbors)

Card ID: 1778592619770

Statement of the bounds on the 1NN Classifier Risk bounds theorem

---

Suppose we're doing binary classification ($k = 2$) and all point distances are distinct ($||x_{(j)} - x||$ unique). Assume the true conditional class probability $p_1(x) = \mathbb{P}(Y = 1 \,|\, X = x)$ is continuous, and X has density bounded away from zero on its support.

Then the misclassification risk $R(h^*)$ fulfills

$$R(h^*) \leq \lim_{n \to \infty} R(h^{1NN}) \leq 2R(h^*)$$

Card ID: 1778592619772

Prop

1NN Classifier Risk
Suppose we're doing binary classification ($k = 2$) and all point distances are distinct ($||x_{(j)} - x||$ unique). Assume the true conditional class probability $p_1(x) = \mathbb{P}(Y = 1 \,|\, X = 1)$ is continuous, and X has density bounded away from zero on its support.
Show that the misclassification risk $R(h^*)$ fulfills

$$R(h^*) \leq \lim_{n\to\infty} R(h^{1NN}) \leq 2R(h^*)$$

First show that $\mathbb{P}(h^{1NN}(x) = 1 \,|\, X_1...X_n) = p_1(X_{(1)}(x))$, and likewise for $k = 2$.

Use this to show $R(h^{1NN}) = \mathbb{E}[p_2(X_{(1)}(X^*))p_1(X^*) + p_1(X_{(1)}(X^*))p_2(X^*)$

Make the approximation that $p_1(X_{(1)}(X^*)) \approx p_1(X^*)$ by continuity.

Plug into $\lim_{n\to\infty} R(h^{1NN})$, recognize that $2\mathbb{E}[p_1(X^*)(1-p_1(X^*))] \leq 2R(h^*)$

Card ID: 1778592619773

3.5 Bagging

Def: Bootstrap Aggregation (Bagging)

Given some low-bias classifier h:
1. For $b = 1...B$, draw n samples $(X_i^{(b)}, Y_i^{(b)})$ WITH REPLACEMENT from observations. Use this to compute the classifier $h^{(b)}$.
2. Create a new compound classifier by majority voting among all bs:

$$h^{bag}(x) \in \operatorname*{argmax}_k \frac{1}{B}\sum_{b=1}^{B} \mathbb{1}_{h^{(b)}(x)=k}$$

or for regression, just an average:

$$\hat{f}^{bag}(x) = \frac{1}{B}\sum_{b=1}^{B} f^{(b)}(x)$$

Card ID: 1778592619774

Benefit of bagging

Won't change the bias, but can reduce variance (because bootstrapped samples are more independent).

Card ID: 1778592619776

Def: m-out-of-n Bagging

and its benefits

Instead of drawing n samples with replacement, draw $m < n$ samples (still with replacement).

This increases independence among bagged samples, further decreasing the variance.

This yields a consistent estimator when $B, m \to \infty$ with $m/n \to 0$.

Card ID: 1778592619777

3.6 Support Vector Machines (SVM) and Kernelization

Def: Support Vector Machine

A two-class classifier ($K = 2$) with points labeled $Y_i \in \{1, -1\}$.
We seek a hyperplane $f(x)$, with $f(x) > 0$ for class 1 and $f(x) < 0$ for class 2 (class -1).
We find the hyperplane fit which maximizes the margin (closest distance from points on each side to the separating hyperplane)

$$h^{SVC}(x) = \text{sgn}(\hat{\alpha} + x^{\text{T}}\hat{\beta})$$

Card ID: 1779308511660

Def: Hyperplane and Separating Hyperplane

$$f(x) = \alpha + x^T\beta$$

It's called separating if

$$Y_i f(X_i) > 0$$

for all $i \in [n]$

Card ID: 1779308511663

Def: Optimal Separating Hyperplane for SVMs

Maximize M (the margin) over $\alpha \in \mathbb{R}$, $\beta \in \mathbb{R}^p$, $M \geq 0$, subject to

$$\frac{Y_i(\alpha + X_i^{\mathrm{T}}\beta)}{||\beta||_2} \geq M$$

for all $i \in [n]$.
Equivalently, we can MINIMIZE $||\beta||_2$ subject to

$$Y_i(\alpha + X_i^{\mathrm{T}}\beta) \geq 1$$

for all $i \in [n]$

Card ID: 1779308511665

Def: Softened Separating Hyperplane Optimization for SVMs

Minimize

$$||\beta||_2^2 + C\sum_{i=1}^{n}\xi_i$$

over $\alpha \in \mathbb{R}$, $\beta \in \mathbb{R}^p$, $\xi \in \mathbb{R}^n$ subject to

$$Y_i(\alpha + X_i^{\mathrm{T}}\beta) \geq 1 - \xi_i$$

$$\xi_i \geq 0$$

for $i \in [n]$. We call the ξ_i slack variables:
If $\xi_i = 0$, X_i is on the right side of the boundary and beyond the margin.
If $0 < \xi_i < 1$, X_i is within the boundary but on the wrong side of the margin.
If $\xi > 1$, then X_i is on the wrong side of the boundary.

A smaller C means a tighter margin with more misclassifications.

Card ID: 1779308511667

Def: Alternative formulation of the soft SVM

$$(\hat{\alpha}, \hat{\beta}) \in \underset{\alpha\in\mathbb{R},\beta\in\mathbb{R}^p}{\operatorname{argmin}} \left(\sum_{i=1}^{n}(1 - Y_i(\alpha + X_i^{\mathrm{T}}\beta))_+ + \lambda||\beta||_2^2\right)$$

Card ID: 1779308511668

Def: Laplace Kernel

$$k(x, y) = \exp(-\gamma||x - y||_2)$$

Radial like the Gaussian kernel, but with a slower decay for far away points.

Card ID: 1779308511670

TODO SLIP version of the representer theorem?

4 Part 3: Machine Learning

4.1 Neural Networks

Def: Sigmoid Function

$$\sigma(x) = \frac{e^x}{(1 + e^x)}$$

Card ID: 1776686107896

Def: ReLU Function

$$\sigma(x) = \max(0, x)$$

Card ID: 1776686107898

Def: Softmax Function

$$g_k(z) = \frac{e^{z_k}}{\sum_{r=1}^{K} e^{z_k}}$$

for $k \in [K]$

Card ID: 1776686107900

Def: Universal Approximation Theorem for Neural Networks

Any (borel-measurable) function can be approximated to a prescribed precision by a single hidden layer neural network with any nonlinear activation function.

Card ID: 1776686107901

Def: Cross-Entropy Loss

$$L(\theta) = -\frac{1}{n}\sum_{i=1}^{n}\sum_{k=1}^{K} Y_{ik}\log P_k^{(\theta)}(X_i)$$

Card ID: 1776686107903

Def: Gradient Descent for Neural Networks

$$\theta^{(t+1)} = \theta^{(t)} - \alpha_t \nabla_\theta \mathrm{Err}_{\mathrm{tr}}(\theta^{(t)})$$

Card ID: 1776686107905

4.2 Decision Trees

Core idea behind decision trees

Break covariate space down into rectangular regions $R_1, R_2, ..., R_M$, each with some contributing factor c_m. When x falls within each, that region contributes its c_m to a final total:

$$x \mapsto \sum_{m=1}^{M} c_m \mathbb{1}_{x \in R_m}$$

The hard part is finding the regions and coefficients that work!

Idea: Start with the entire space in one big rectangle, then split it into two smaller pieces, staying axis-aligned:

$$R_l(j,t) = \{x \in R : x_j \leq t\}$$

$$R_r(j,t) = \{x \in R : x_j > t\}$$

where $j \in [p]$ are the split variables and $t \in \mathbb{R}$ are the split points (we choose only one variable, and one location along that variable, to split over at a time)

Find the best split variable and split point to minimize a cost function $Q(j,t)$.

Card ID: 1775594408226

Explain the general algorithm for a regression tree

Given data $(x_i, y_i)_{i\in[n]}$ in $\mathbb{R}^p \times \mathbb{R}$, continually split any region so as long as it has at least 2 data points.

Let T_j be the set of midpoints between adjacent points in a region $x_{ij} \in \mathbb{R}$. Compute the cost-minimizing split to decide where to split:

$$(\hat{j}_R, \hat{t}_R) \in \underset{j\in[p],\, t\in T_j}{\operatorname{argmin}} \; Q(j,t)$$

where $Q(j,t)$ is the loss after each split.

Then perform the split and continue!

Card ID: 1775594408228

Def: Regression decision tree loss

It's just the best case loss of the left rectangle (best choice of c_l) plus the best case loss of the right rectangle (best choice of c_r) minus the current loss of the rectangle (using c):

$$\begin{aligned} Q_{reg}(j,t) = & \min_{c_l\in\mathbb{R}} \sum_{i:x_i\in R_l(j,t)} (y_i - c_l)^2 = \sum_{i:x_i\in R_l(j,t)} (y_i - c(R_l(j,t))^2 \\ & + \min_{c_r\in\mathbb{R}} \sum_{i:x_i\in R_r(j,t)} (y_i - c_r)^2 + \sum_{i:x_i\in R_r(j,t)} (y_i - c(R_r(j,t))^2 \\ & - \min_{c\in\mathbb{R}} \sum_{i:x_i\in\mathbb{R}} (y_i - c)^2 + \sum_{i:x_i\in R} (y_i - c(R))^2 \end{aligned}$$

where

$$c(R) = \frac{1}{N(R)} \sum_{i:x_i\in\mathbb{R}} y_i$$

where $N(R)$ is the number of points in a particular region.

Note that this can never be positive! Splitting will never make things worse (because we could just have two splits with the same old height c).

Card ID: 1775594408230

Final regression function of a regression tree

$$\hat{f}_{CART}(x) = \sum_{m=1}^{M} \hat{c}_m \mathbb{1}_{x \in R_m}$$

with $\hat{c}_m = c(R_m)$, where

$$c(R) = \frac{1}{N(R)} \sum_{i: x_i \in \mathbb{R}} y_i$$

where $N(R)$ is the number of points in a particular region.

R_m are called the leaves of the decision tree!

Card ID: 1775594408232

Heuristic for when a regression tree is done growing

Each region should contain no more than 5 points

Card ID: 1775594408234

Runtime of growing a regression tree (when optimized)

O(n)

Card ID: 1775594408235

TODO HOW TO MAKE CART FAST?

Def: Gini Index

$$G(R) := \sum_{k=1}^{K} \hat{p}_k(R)(1 - \hat{p}_k(R))$$

where

$$\hat{p}_k(R) := \frac{1}{N(R)} \sum_{i:X_i \in R} I_{Y_i = k}$$

(fraction of points in R that correspond to class k)

Measures the impurity of a region R: we want regions with a lot of points with the same label.
(pick a label randomly from the pool of labels we know the points have; what's the misclassification rate of just picking randomly from those labels, to label each point?)

Card ID: 1778592619787

Def: Classification Tree Loss

For classification trees, we use the loss:

$$Q(j,t) := \frac{N(R_l(j,t))}{N(R)} G(R_l(j,t)) + \frac{N(R_r(j,t))}{N(R)} G(R_r(j,t)) - G(R)$$

where $G(R)$ is the Gini index.

(Reduction in impurity that splitting gains, weighted by the fraction of points that each region takes)

Card ID: 1778592619788

Def: Final classification function of a classification tree

$$h^{CART}(x) := \sum_{m=1}^{M} \hat{c}_m I_{x \in R_m}$$

where

$$\hat{c}_m = \operatorname*{argmax}_k \hat{p}_k(R_m)$$

Card ID: 1778592619790

Def: Bias-Variance Tradeoff for CART

Shallow trees: high bias, low variance
Deep trees: low bias, high variance

Card ID: 1778592619793

4.3 Random Forests

Def: Random Forest Procedure

Similar to bagging:
1. For $b = 1...B$, draw n samples $(X_i^{(b)}, Y_i^{(b)})$ WITH REPLACEMENT. At each split, pick the best out of some random selection m_{try} of the p variables. Continue until it has grown maximally.
2. Aggregate all B tree functions. For regression, just take the average of applying each tree function; for classification, just take the majority vote.

Card ID: 1778592619796

Variance of a regression random forest

$$\text{Var}[f^{RF}(x)] = \rho\sigma^2 + \frac{1-\rho}{B}\sigma^2$$

where for $b \neq b'$, $\sigma^2 = \text{Var}[f^{(b)}(x)]$ and ρ is the correlation between $f^{(b)}(x)$ and $f^{(b')}(x)$.

(split correlation into diagonal and non-diagonal bits in the sum)

Card ID: 1778592619798

Def: Out-of-bag Error (OOB)

Just the average difference between Y_i and an aggregated forest prediction $\hat{Y}_i$. Estimates the prediction error.

Note this approaches leave-one-out cross validation error as $B \to \infty$.

Card ID: 1778592619800

5 Part 4: Unsupervised Learning

Def: Define a unitary matrix

A matrix A where $A^* = A^{-1}$,
where A^* is the complex conjugate

For real matricies this is the same as being orthogonal

Card ID: 1775594408237

5.1 Principal Component Analysis

Def: Define an orthogonal matrix, what do they represent?

A matrix A where $A^{\mathrm{T}} = A^{-1}$

it represents rotations and reflections (but not stretching)

Card ID: 1775594408239

Def: Thin Singular Value Decomposition

Given $X \in \mathbb{R}^{n \times p}$, the decomposition

$$X = UDV^{\mathrm{T}}$$

where $U \in \mathbb{R}^{n \times p}$ has orthonormal columns (ie fulfills $U^{\mathrm{T}}U = I$),
$D \in \mathbb{R}^{p \times p}$ is a sorted diagonal matrix ($D_{11} \geq D_{22} \geq ...$),
and $V \in \mathbb{R}^{p \times p}$ is also unitary.

This can be thought of as a process where a space is rotated/reflected, stretched on its axes, then rotated/reflected again.

Card ID: 1775594408241

Def: Principal Component

Given some $X \in \mathbb{R}^{n \times p}$ row-centered ($X^T 1 = 0$), PCA finds a sequence of p unit vectors v_j, where each v_j MAXIMIZES the sample variance $\mathrm{Var}(Xv_j) = \frac{1}{n} v^{\mathrm{T}} X^{\mathrm{T}} X v$ while staying orthogonal to all previous v_is.

We call Xv_j the principal components.

Card ID: 1775594408242

State how a thin singular value decomposition yields the principal components

---

Given a singular value decomposition UDV, substituting into the sample variance $v^{\mathrm{T}} X^{\mathrm{T}} X v$ and optimizing tells us that $v_j = V_j$ (jth column of V) and so for the principal component, we have:

$$Xv_j = XV_j = D_{jj} U_j$$

where U_j is the jth row of U.

Card ID: 1775594408244

In practice, how many principal components do we use?

Take the smallest $J \leq P$ such that

$$\frac{\sum_{j=1}^{J} D_{jj}^2}{\sum_{j=1}^{p} D_{jj}^2} \geq 0.8$$

Card ID: 1775594408245

5.2 K-Means Clustering

Goal of K-Means Clustering

Find a 'good enough' labelling minimizing

$$L(c_1 ... c_k) = \sum_{k=1}^{K} \frac{1}{2|C_k|} \sum_{i \in C_k} \sum_{j \in C_k} ||x_i - x_j||_2^2$$

(minimize within-cluster variation)
But this is NP-hard! Need a heuristic solution.
Card ID: 1775594408247

Def: Centroid of a cluster

$$\bar{X}_k = \frac{1}{|C_k|} \sum_{i \in C_k} X_i$$

(center of mass)

Note that we can represent the within-cluster variance loss in terms of this:

$$L(c_1 ... c_k) = \sum_{k=1}^{K} \sum_{i \in C_k} ||x_i - \bar{x}_k||_2^2$$

Card ID: 1775594408248

Def: K-Means Clustering Algorithm

1. Initialize a K-partition $C_1 ... C_k$ randomly.

2. Until convergence:

2-1. Compute the centroid of each cluster $\hat{x}_i$

2-2. Construct a new partition $\tilde{C}_1 ... \tilde{C}_k$ by assigning each point to the closest centroid.

Note that this only finds a local minimum, not always the global minimum!

Card ID: 1775594408249

K-Means best practices (how do we choose K?)

To choose K, try a bunch and see which ones offer the most explainability. (don't just increase to minimize loss; $K = n$ always does that)

Good clusterings should be stable with repeated runs, or with the addition or removal of a few data points.

Card ID: 1775594408250

6 Appendix: Practicals

6.1 Practical 1

We spoke about the weird notation for regression in R: -1 specifies no intercept; $x : z$ specifies an interaction term (xz), $x * z$ specifies all interactions.

Def: Residual Squared Error

$$RSE = \sqrt{\frac{\sum_{i=1}^{n}(y_i - \hat{y}_i)^2}{n - p}}$$

where p is the number of parameters in our regression (denominator is the number of degrees of freedom).

Card ID: 1770113248226

Def: R^2 (Coefficient of determination)

$$R^2 = 1 - \frac{RSS}{TSS} = 1 - \frac{\sum_i (y_i - \hat{y}_i)^2}{\sum_i (y_i - \bar{y})^2}$$

RSS is the residual sum of squares, and TSS is the total sum of squares. Proportion of the variation predictable from our model. We want this to be large!

Card ID: 1770113248238

Def: Cook's Distance, Intuitively

A datapoint (x, y)'s cook's distance is how much it would affect the final fit if it were to be removed.

Card ID: 1778592619810