Causal Inference
CS 477-677

Estimator Reliability And Regression Models

Ilya Shpitser

Outline

1. The Bootstrap
2. Linear Regression
3. Logistic Regression
4. Overfitting and Regularization

Reliability Of An Estimator

- Recall the picture: $p(\vec{x})$ generates $\vec{X}_{n\times k}$.
- We want to estimate some parameter θ of $p(\vec{x})$ using a rule (our estimator) that maps $\vec{X}_{n\times k}$ to $\hat{\theta}(\vec{X}_{n\times k})$.
- Say this rule does a good job on a particular $\vec{X}_{n\times k}$.
- How do we know this rule works well for other $\vec{X}_{n\times k}$?
- Most $\vec{X}_{n\times k}$?
- All $\vec{X}_{n\times k}$?
- Want to measure reliability of our rule in this sense.

How To Measure Reliability?

- Say we knew $p(\vec{x})$.
- We could sample many many $\vec{X}^1_{n\times k}, \vec{X}^2_{n\times k}, \ldots, \vec{X}^m_{n\times k}$.
- See if estimator does well with $\hat{\theta}(\vec{X}^1_{n\times k})$, $\hat{\theta}(\vec{X}^2_{n\times k}), \ldots, \hat{\theta}(\vec{X}^m_{n\times k})$.
- If so, estimator is reliable.

How To Measure Reliability?

- Say we knew $p(\vec{x})$.
- We could sample many many $\vec{X}^1_{n\times k}, \vec{X}^2_{n\times k}, \ldots, \vec{X}^m_{n\times k}$.
- See if estimator does well with $\hat{\theta}(\vec{X}^1_{n\times k})$, $\hat{\theta}(\vec{X}^2_{n\times k}), \ldots, \hat{\theta}(\vec{X}^m_{n\times k})$.
- If so, estimator is reliable.
- But if we know $p(\vec{x})$, we already know θ!
- All we have is a single $\vec{X}_{n\times k}$ that (we think..) is drawn from $p(\vec{x})$.
- What do we do?

Key Ideas For Measuring Reliability

- We don't know $p(\vec{x})$, but we can view $\vec{X}_{n \times k}$ is an approximation if viewed as a histogram.

Key Ideas For Measuring Reliability

- We don't know $p(\vec{x})$, but we can view $\vec{X}_{n \times k}$ is an approximation if viewed as a histogram.

On the board

Key Ideas For Measuring Reliability

- We don't know $p(\vec{x})$, but we can view $\vec{X}_{n\times k}$ is an approximation if viewed as a histogram.
- View $\vec{X}_{n\times k}$ as an empirical distribution, each row x_{i*} has probability $1/n$.
- Generate many many $\vec{X}^1_{n\times k}, \vec{X}^2_{n\times k}, \ldots, \vec{X}^m_{n\times k}$ using this distribution.

Key Ideas For Measuring Reliability

- We don't know $p(\vec{x})$, but we can view $\vec{X}_{n\times k}$ is an approximation if viewed as a histogram.
- View $\vec{X}_{n\times k}$ as an empirical distribution, each row x_{i*} has probability $1/n$.
- Generate many many $\vec{X}^1_{n\times k}, \vec{X}^2_{n\times k}, \ldots, \vec{X}^m_{n\times k}$ using this distribution.
- See how $\hat{\theta}(\vec{X}_{n\times k}) - \hat{\theta}(\vec{X}^i_{n\times k})$ for $i = 1, \ldots, m$ behaves.

Key Ideas For Measuring Reliability

- We don't know $p(\vec{x})$, but we can view $\vec{X}_{n\times k}$ is an approximation if viewed as a histogram.
- View $\vec{X}_{n\times k}$ as an empirical distribution, each row x_{i*} has probability $1/n$.
- Generate many many $\vec{X}^1_{n\times k}, \vec{X}^2_{n\times k}, \ldots, \vec{X}^m_{n\times k}$ using this distribution.
- See how $\hat{\theta}(\vec{X}_{n\times k}) - \hat{\theta}(\vec{X}^i_{n\times k})$ for $i = 1, \ldots, m$ behaves.
- Note: we do not know θ or $p(\vec{x})$. We are using $\hat{\theta}(\vec{X}_{n\times k})$ as an approximation of θ, and $\vec{X}_{n\times k}$ as an approximation of $p(\vec{x})$.

The Bootstrap

Given dataset $\vec{X}_{n\times k}$,

- Generate $\vec{X}^1_{n\times k}, \vec{X}^2_{n\times k}, \ldots, \vec{X}^m_{n\times k}$ by sampling rows from $\vec{X}_{n\times k}$ with replacement.
- Estimate and store $\hat{\theta}(\vec{X}_{n\times k}) - \hat{\theta}(\vec{X}^i_{n\times k})$ for $i = 1, \ldots, m$.
- Look at (for example) 2.5% and 97.5% quantiles of the resulting histogram, call them Q_l, Q_u.

The Bootstrap

Given dataset $\vec{X}_{n\times k}$,

- Generate $\vec{X}^1_{n\times k}, \vec{X}^2_{n\times k}, \ldots, \vec{X}^m_{n\times k}$ by sampling rows from $\vec{X}_{n\times k}$ with replacement.
- Estimate and store $\hat{\theta}(\vec{X}_{n\times k}) - \hat{\theta}(\vec{X}^i_{n\times k})$ for $i = 1, \ldots, m$.
- Look at (for example) 2.5% and 97.5% quantiles of the resulting histogram, call them Q_l, Q_u.

On the board

The Bootstrap

Given dataset $\vec{X}_{n\times k}$,

- Generate $\vec{X}^1_{n\times k}, \vec{X}^2_{n\times k}, \ldots, \vec{X}^m_{n\times k}$ by sampling rows from $\vec{X}_{n\times k}$ with replacement.
- Estimate and store $\hat{\theta}(\vec{X}_{n\times k}) - \hat{\theta}(\vec{X}^i_{n\times k})$ for $i = 1, \ldots, m$.
- Look at (for example) 2.5% and 97.5% quantiles of the resulting histogram, call them Q_l, Q_u.
- Claim: if we do this procedure lots of times, interval $[\hat{\theta}(\vec{X}_{n\times k}) + Q_l, \hat{\theta}(\vec{X}_{n\times k}) + Q_u]$ will contain $\hat{\theta}(\vec{X}_{n\times k})$ 95% of the time.
- This is called a confidence interval.
- In practice, report $\hat{\theta}(\vec{X}_{n\times k})$ and confidence interval as a measure of reliability of procedure (not the estimate!).
- Bayesians report quantiles of the posterior distribution (more later).

Why Learn The Bootstrap?

- Works for many (but by no means all) θ.
- For some θ can calculate confidence intervals in closed form using algebra.
- Can't do that in this class – many θ will be complicated.
- Bootstrap does need certain conditions to work, will skip details.
- Can evaluate just using observed data – similar to cross-validation in machine learning.

Regression Models

- Regression models relate a set of *features* $\vec{X}$ and an outcome Y.
- Useful for predicting the mean of Y, classification, etc.
- We will consider two fairly simple regression models in this class.
- Large literature in statistics and ML on regression models, they could get very complex.
- Lots of other models in ML – support vector machines, decision trees, neural networks, etc.
- Will leave those aside in this class.

Linear Regression

- Data: n (continuous) realizations of $\vec{X} \cup \{Y\}$, written as $\mathcal{D}_{n\times(k+1)}$, which is $\vec{Y}_{n\times 1}$ and $\vec{X}_{n\times k}$.
- Model ($\epsilon \sim \mathcal{N}(0, \sigma^2)$, independent of X):

$$Y = a_0 + \sum_{i=1}^{k} x_i \cdot a_i + \epsilon$$

- Likelihood (for Y conditional on $\vec{X}$):

$$\mathcal{L}_{Y|\vec{X}}(\mathcal{D}; \{(a_{1i}), \sigma^2\}) = \prod_{i=1}^{n} \frac{1}{\sqrt{2\sigma^2\pi}} \exp\left\{-\frac{(y_i - (a_0 + \sum_{j=1}^{k} a_j \cdot x_{ij}))^2}{2\sigma^2}\right\}$$

- Log Likelihood:

$$\mathcal{L}_{Y|\vec{X}}(\mathcal{D}; \{(a_{1i}), \sigma^2\}) = \sum_{i=1}^{n} -\frac{(y_i - (a_0 + \sum_{j=1}^{k} a_j \cdot x_{ij}))^2}{2\sigma^2} - n \log\left\{\sqrt{2\sigma^2\pi}\right\}$$

- Equivalent to minimizing $\sum_{i=1}^{n}(y_i - (a_0 + \sum_{j=1}^{k} a_j \cdot x_{ij}))^2$ (sometimes named "least squares").

Maximizing Likelihood

- Concisely in matrix form ($\tilde{X}_{n\times(k+1)}$ has first column of 1s):

$$Y_{n\times 1} = \tilde{X}_{n\times(k+1)} \cdot A_{(k+1)\times 1} + e_{n\times 1}$$

- Then we are minimizing:

$$(Y - \tilde{X} \cdot A)^T_{1\times n} \cdot (Y - \tilde{X} \cdot A)_{n\times 1} = e^T \cdot e$$

- Derivation:

$$\begin{aligned}(Y - \tilde{X} \cdot A)^T \cdot (Y - \tilde{X} \cdot A) &= (Y^T - A^T \cdot \tilde{X}^T) \cdot (Y - \tilde{X} \cdot A) \\ &= Y^T Y - Y^T \tilde{X} A - A^T \tilde{X}^T Y + A^T \tilde{X}^T X A \\ &= Y^T Y - 2Y^T \tilde{X} A + A^T \tilde{X}^T \tilde{X} A\end{aligned}$$

- Differentiate (wrt A) and set to 0:

$$\begin{aligned}0 &= -2\tilde{X}^T Y + 2\tilde{X}^T X A \\ \tilde{X}^T \tilde{X} A &= \tilde{X}^T Y \\ A &= (\tilde{X}^T \tilde{X})^{-1} \tilde{X}^T Y\end{aligned}$$

Polynomial Regression

- For a single X, $Y = a + bX$ is a line.
- But may want a higher order polynomial (Y is height of artillery shell, X is time...), then $Y = a + bX + cX^2$.
- Y and X have a non-linear relationship, but problem still linear with respect to parameters!
- Can use same machinery, just add a column for X^2 to your data!

Interactions

- Features can affect outcomes in synergistic ways, not just individually.

Interactions

- Features can affect outcomes in synergistic ways, not just individually.
- Examples: coca cola + mentos, alcohol + prescription painkiller.
- Simple to modify regression models to handle this:

$$Y = w_0 + w_1 x_1 + w_2 x_2 + w_{12} x_1 x_2 + \epsilon.$$

- Same machinery as before.

Interactions

- Features can affect outcomes in synergistic ways, not just individually.
- Examples: coca cola + mentos, alcohol + prescription painkiller.
- Simple to modify regression models to handle this:

$$Y = w_0 + w_1 x_1 + w_2 x_2 + w_{12} x_1 x_2 + \epsilon.$$

- Same machinery as before.
- w_1, w_2 are known as main effects.
- w_{12} is known as an interaction effect.
- In practice, interaction effects are often smaller than main effects ("sparsity of effects principle").

Logistic Regression

- Data: n (continuous or discrete) realizations of $\vec{X}$, binary realization of Y, written as $D_{n\times(k+1)}$ or $\vec{Y}_{n\times 1}$ and $\vec{X}_{n\times k}$ as before.
- Model:

$$p(Y = 1 \mid \vec{x}; \vec{w}) = \frac{1}{1 + \exp\left\{-\sum_{i=1}^{k} x_i \cdot w_i\right\}}.$$

- Likelihood (for Y conditional on $\vec{X}$):

$$\mathcal{L}_{Y|\vec{X}}(\mathcal{D}; \{w_1, \ldots w_k\}) = \left(\prod_{j:y_j=1} \frac{1}{1 + \exp\left\{-\sum_{i=1}^{k} x_{ji} \cdot w_i\right\}}\right) \cdot \left(\prod_{j:y_j=0} 1 - \frac{1}{1 + \exp\left\{-\sum_{i=1}^{k} x_{ji} \cdot w_i\right\}}\right)$$

Logistic Regression (cont.)

- Log Likelihood:

$$\log \mathcal{L}_{Y|\vec{X}}(D; \{w_1, \ldots w_k\}) = -\sum_{j:y_j=1} \log\left(1 + \exp\left\{-\sum_{i=1}^{k} x_{ji} \cdot w_i\right\}\right) + \sum_{j:y_j=0} \log\left(1 + \exp\left\{-\sum_{i=1}^{k} x_{ji} \cdot w_i\right\}\right)$$

- Can show (board):

$$\frac{\partial \log \mathcal{L}_{Y|\vec{X}}(D; \vec{w})}{\partial \vec{w}} = \sum_{j=1}^{n} x_{j(1:k)}(y_j - p(Y = 1 \mid x_{j(1:k)}))$$

- Setting this to 0 yields transcendental equations.

Interlude: Semi-Parametric Models

- Regression models so far were parametric models of $p(Y \mid \vec{X})$.
- Alternative view: semi-parametric models of $p(Y, \vec{X})$.

Interlude: Semi-Parametric Models

- Regression models so far were parametric models of $p(Y \mid \vec{X})$.
- Alternative view: semi-parametric models of $p(Y, \vec{X})$.
- Restricted moment model:

$$Y = \mu(\vec{X}; \vec{w}) + \epsilon; E[\epsilon \mid \vec{X}] = 0.$$

- Mean is parametric, noise is non-parametric: $p(\epsilon \mid \vec{X})$ can be almost anything, provided the expected value is 0!

Interlude: Semi-Parametric Models

- Regression models so far were parametric models of $p(Y \mid \vec{X})$.
- Alternative view: semi-parametric models of $p(Y, \vec{X})$.
- Restricted moment model:

$$Y = \mu(\vec{X}; \vec{w}) + \epsilon; E[\epsilon \mid \vec{X}] = 0.$$

- Mean is parametric, noise is non-parametric: $p(\epsilon \mid \vec{X})$ can be almost anything, provided the expected value is 0!
- Logistic regression is in this class (why?)

Interlude: Semi-Parametric Models

- Regression models so far were parametric models of $p(Y \mid \vec{X})$.
- Alternative view: semi-parametric models of $p(Y, \vec{X})$.
- Restricted moment model:

$$Y = \mu(\vec{X}; \vec{w}) + \epsilon; E[\epsilon \mid \vec{X}] = 0.$$

- Mean is parametric, noise is non-parametric: $p(\epsilon \mid \vec{X})$ can be almost anything, provided the expected value is 0!
- Logistic regression is in this class (why?)
- In semi-parametric models we will often think of a vector of parameters (β, η) where β is a finite-dimensional vector of target parameters, and η is an infinite-dimensional vector of nuisance parameters.
- Here $\beta = \vec{w}$ is of interest, and whatever crazy parameters $p(\epsilon \mid \vec{X})$ and $p(\vec{X})$ have are nuisance.

Inference In Semi-Parametric Models

- We want to make inferences on β. Want to construct regular, asymptotically linear (RAL) estimators, that look like:

$$n^{1/2}(\hat{\beta}(\vec{X}_{n\times k}) - \beta) = n^{-1/2}\sum_{i=1}^{n}\phi(\vec{x}_{i(1:k)}) + o_p(1).$$

- Asymptotically linear: looks like a sum. Regular (loosely): does the same algorithm everywhere, does not special case any part of the parameter space.
- β are true parameters, $o_p(1)$ is something tiny that goes to zero, and $\phi(.)$ is an object called an influence function.
- $\phi(.)$ tells you how much each data row influences the final answer. Also tells us the asymptotic variance of estimator since:

$$n^{1/2}(\hat{\beta}(\vec{X}_{n\times k})) - \beta) \xrightarrow{\mathcal{D}} \mathcal{N}(0, E[\phi\phi^T]).$$

- If we learn $\phi(.)$ for a given β in some semi-parametric model, we are done. May discuss deriving $\phi(.)$ later...

Influence Functions For The Logistic Model

- Can be shown (notes later) that the class of RAL estimators for $\beta = \vec{w}$ of the semi-parametric logistic regression model is:

$$\sum_{i=1}^{n} A(\vec{x}_{i(1:k)})\{y_i - \mu(\vec{x}_{i(1:k)}; \vec{w})\} = 0; A(\vec{x}) \text{ any function of } \vec{x}.$$

- Note that the MLE is in this class, with $A(\vec{x}) = \vec{x}$.
- In fact is the most efficient choice for $A(\vec{x})$.
- Can solve these equations by the Newton-Raphson method.

Newton Method

- Say we want to find the minimum of a smooth f.
- Can approximate f "near" a point x_0, using Taylor expansion:

$$f(x) = f(x_0) + (x - x_0) \cdot f'(x_0) + \frac{1}{2} f''(x_0)(x - x_0)^2 + \dots.$$

- So:

$$\frac{\partial f(x)}{\partial x} \approx f'(x_0) + f''(x_0)(x - x_0) \text{ and so}$$

$$x = x_0 - \frac{f'(x_0)}{f''(x_0)}$$

- Approximated expansion so approximate solution – get a new x. Iterate this! (board picture).
- If $x = x_0$, the process stops.
- Can show that if we start close enough to the true x_0, will rapidly converge to it.

Multivariate Newton Method

- Logistic regression has lots of parameters, so need $f(\vec{x})$, for a set $\vec{x}$.
- Can define multivariate version of Taylor's expansion (will skip this).
- Will need a *vector* (Jacobian) of derivatives

$$\nabla f(\vec{x}) = (f'(\vec{x})_{x_1}, f'(\vec{x})_{x_2}, \dots f'(\vec{x})_{x_k})$$

- and a *matrix* (Hessian) of second derivatives:

$$H(\vec{x}) = \begin{pmatrix} f''(\vec{x})_{x_1,x_1} & f''(\vec{x})_{x_1,x_2} & \dots & f''(\vec{x})_{x_1,x_k} \\ \dots & \dots & \dots & \dots \\ f''(\vec{x})_{x_k,x_1} & f''(\vec{x})_{x_k,x_2} & \dots & f''(\vec{x})_{x_k,x_k} \end{pmatrix}$$

- Update rule:

$$\vec{x}^{(m+1)} = \vec{x}^{(m)} - H(\vec{x}^{(m)})^{-1} \cdot \nabla f(\vec{x}^{(m)})$$

First and Second Derivatives for Logistic Regression

- Already have:

$$\frac{\partial \log \mathcal{L}}{\partial \vec{w}} = \sum_{i=1}^{n} x_{i(1:k)}(y_i - p(Y = 1 \mid x_{i(1:k)}))$$

- Can similarly show (board):

$$\frac{\partial \log \mathcal{L}}{\partial \vec{w} \partial \vec{w}^T} = -\sum_{i=1}^{n} x_{i(1:k)}(x_{i(1:k)})^T p(Y = 1 \mid x_{i(1:k)})(1-p(Y = 1 \mid x_{i(1:k)}))$$

- Both linear and logistic model fitting implemented as a part of `glm` function in R.

Fitting Probabilistic Models and Causal Inference

- In ML and Statistics, probabilistic models are used for a wide variety of problems – classification, density estimation, reinforcement learning, etc.
- Will use them in causal inference as part of a broader framework linking counterfactual with factual.
- Need statistical models to make good use of factual data...
- But how do we interpret an analysis causally, in an appropriate way?
- Need a causal model. Will start with the simplest one next time.

Next time: Counterfactuals And Randomization Based Inference