



DS-UA 112

Introduction to Data Science

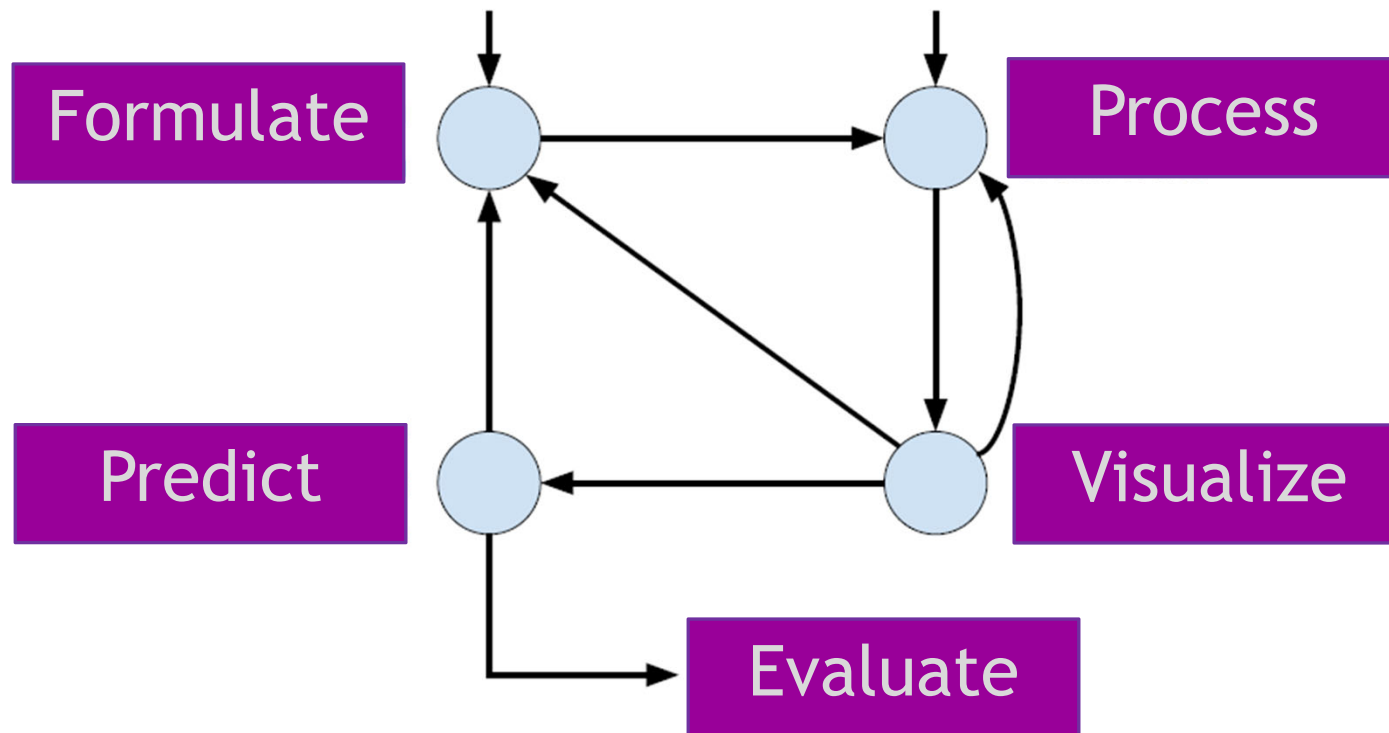
Final Exam Review



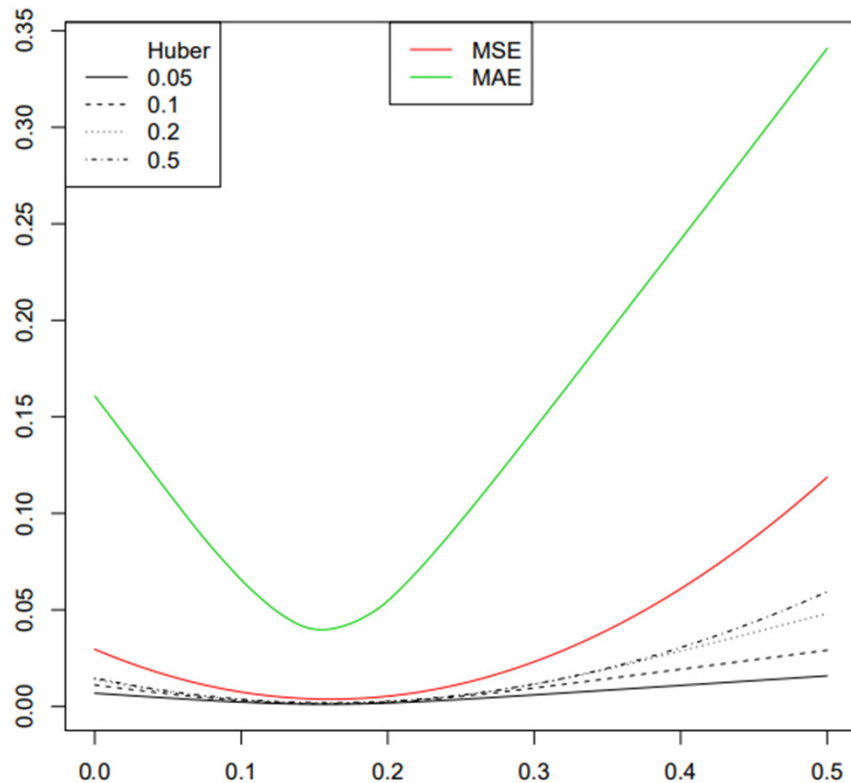
Announcements

- ▶ The exam will be open note and open textbook. However you cannot use search engines.
- ▶ You can access the exam on Gradescope
 - ▶ The exam will be available for 24 hours starting 6:00PM EST on Wednesday May 13
 - ▶ You have a 120 minute time limit.
- ▶ The exam focuses on the second half of the semester. Questions will involve true/false, multiple choice, short answer and calculations

Steps for Data Science



Loss Functions



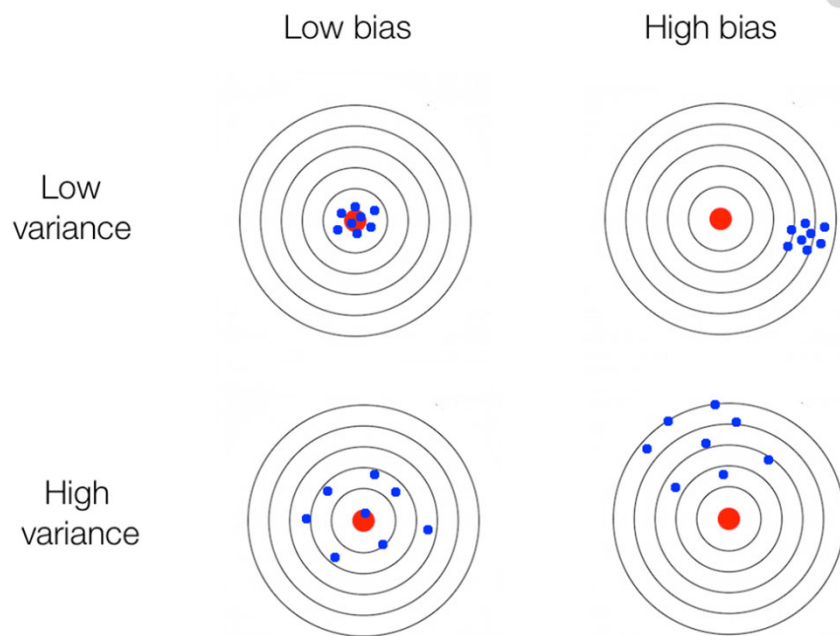
- ▶ The **mean square error** has derivatives at all values of the function. Derivatives are helpful for finding minimum values.
- ▶ However, the mean square error has large output for large input. The function is not **robust** to outliers
- ▶ While the **mean absolute error** has a tricky derivative, the function does not have the same problem with outliers
- ▶ **Huber Loss** combines benefits of both loss functions

Risk

For example the square loss is
 $L(\theta, X) = (\theta - X)^2$

- ▶ Suppose we have a loss function L depending on dataset x_1, \dots, x_n and unknown quantity θ
- ▶ For fixed value of θ , we can compare the value of $L(\theta, x_1, \dots, x_n)$ across different datasets
- ▶ We can take the datasets to correspond to different values of a random variable. If we repeatedly observe n values of a random variable X , then we can compute $L(\theta, x_1, \dots, x_n)$ for each dataset
- ▶ **Risk** is the expectation of a loss function for random variable
 $E[L(\theta, X)]$
- ▶ We need to know the probability distribution of X to compute the expectation.
- ▶ If we can compute the expectation, then we better understand the value of the loss function across different random samples

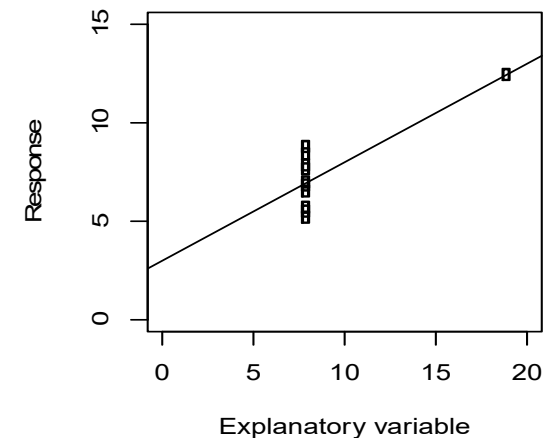
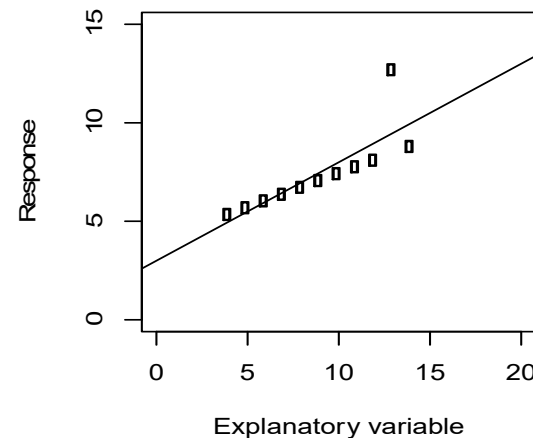
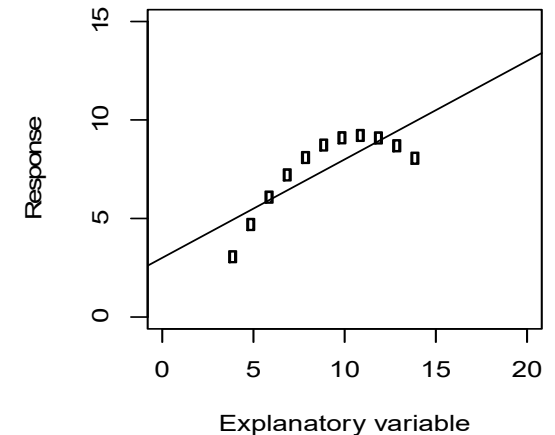
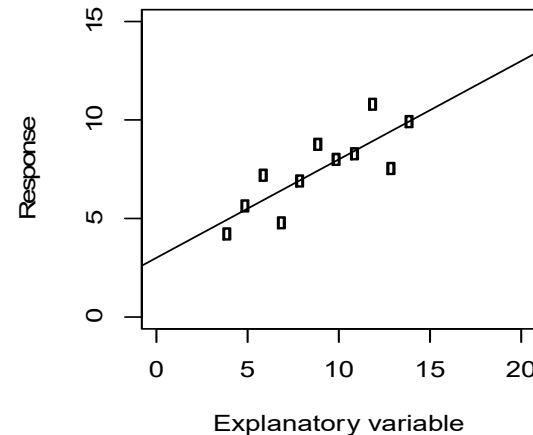
Bias and Variance



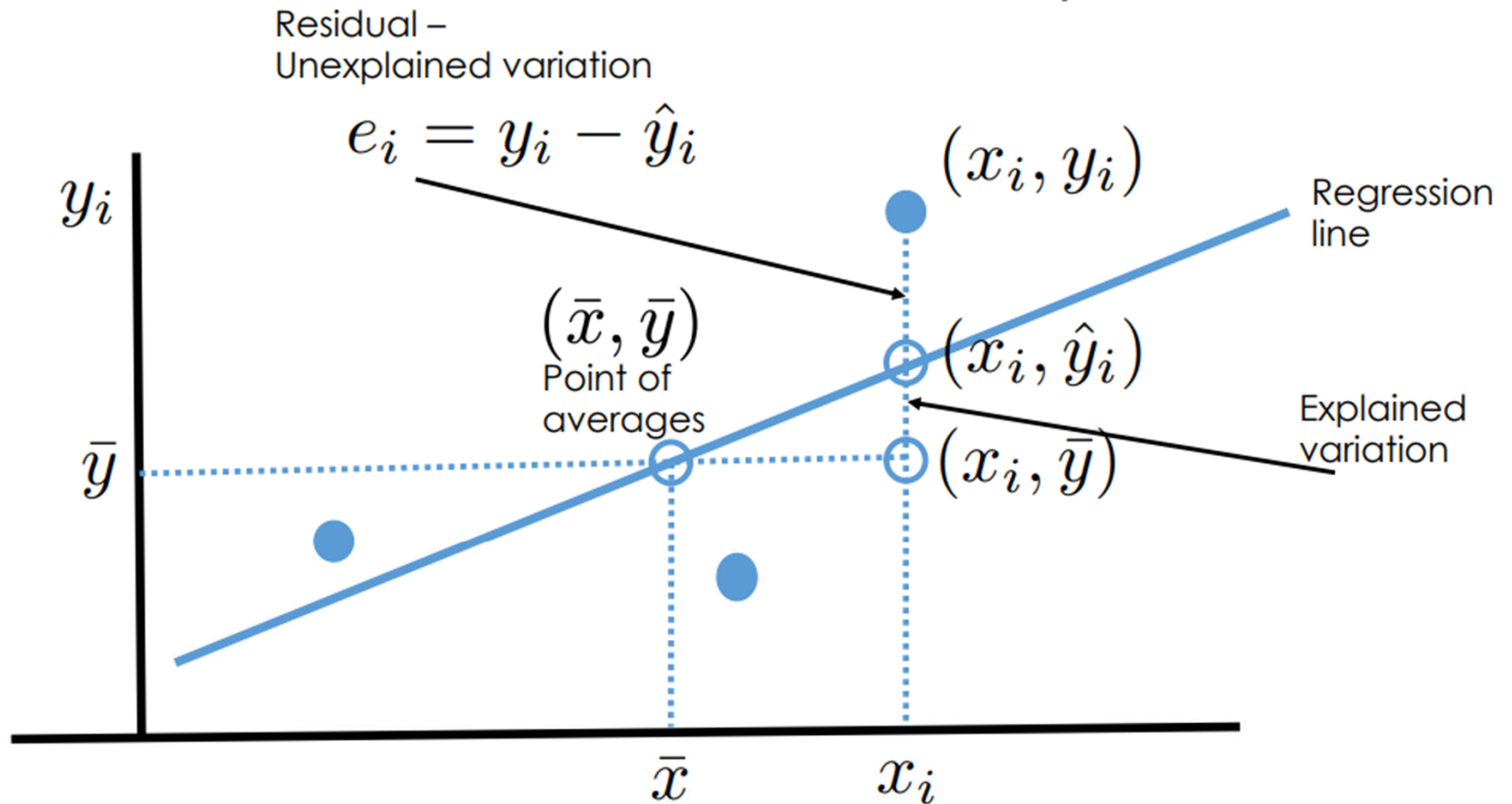
- ▶ We want to choose θ to make $E[L(\theta, X)]$ small. The choice of θ that makes the expectation smallest is $\hat{\theta}$. We can write $\hat{\theta}_n$ to remind us that the estimate from n samples.
- ▶ For the square loss, we can break the risk into two components
 - ▶ Bias measuring the accuracy of the estimator
 - ▶ Variance measuring the consistency of the estimator
- ▶ Here bias does not refer to a property of data but to a tendency of estimators

Correlation

- ▶ Correlation measures the concentration around a line in a scatter-plot of the independent and dependent variables.
- ▶ Correlation is a number between -1 and 1. Around 1 or -1 we have strong positive or negative correlation. Around 0 we have no correlation.
- ▶ Correlation measures association between variables. We cannot measure causation.

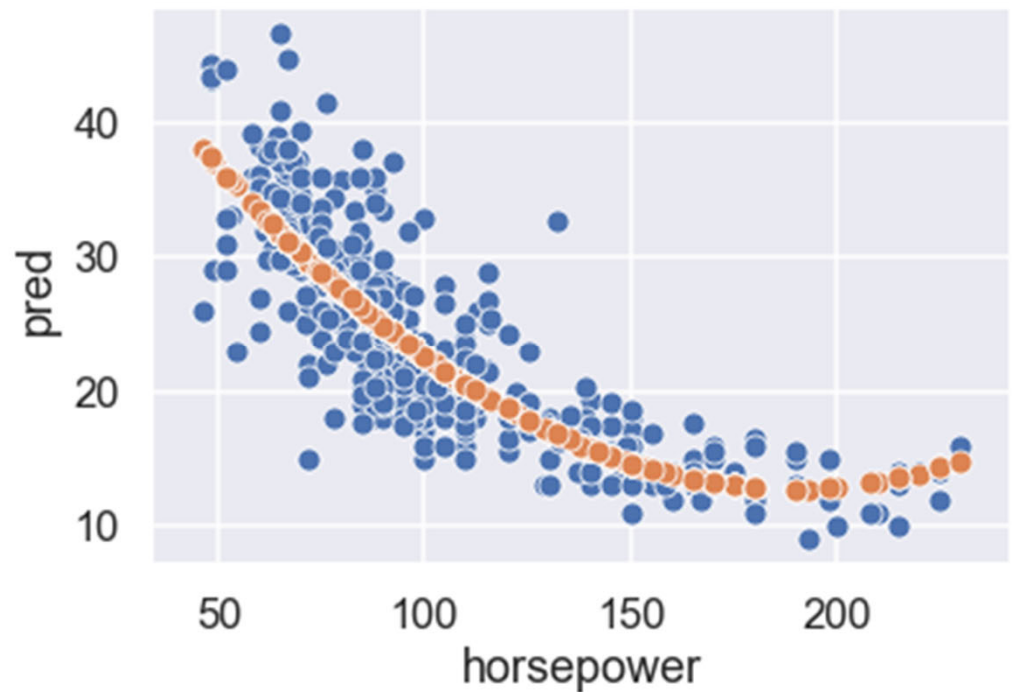


Linear Regression



Polynomial Transformations

- ▶ If we replace an independent variable x with powers $1, x, x^2, x^3, \dots$ then we have a polynomial transformation
- ▶ If we have multiple independent variables then we can multiply them to model interactions between the features.



One-Hot Encoding

- ▶ If we have qualitative data, then we must transform it to quantitative data. However we should be careful with the **encoding** of the categories.
- ▶ We can add another independent variable for each category. The additional variables take the value 0 or 1. We call it a **one-hot encoding**

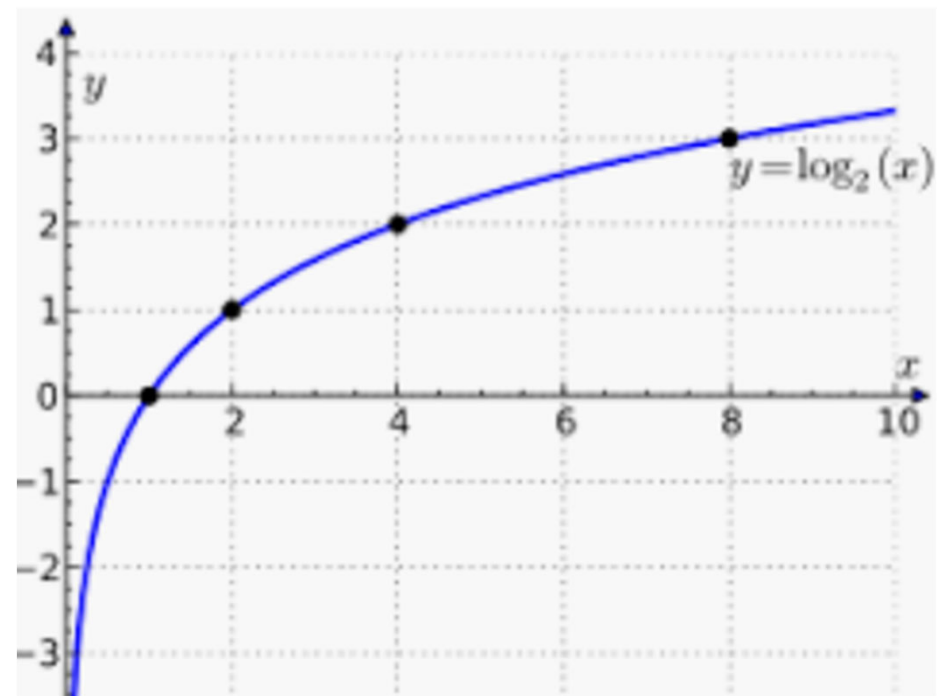
origin	origin=usa	origin=europe	origin=japan
usa	1	0	0
usa	1	0	0
europe	0	1	0
...
usa	1	0	0
japan	0	0	1
japan	0	0	1

Logarithmic Transformations

- Remember that logarithmic transformations help us with visualization. We can transform a large range of numbers to a small range of numbers suitable for a chart.

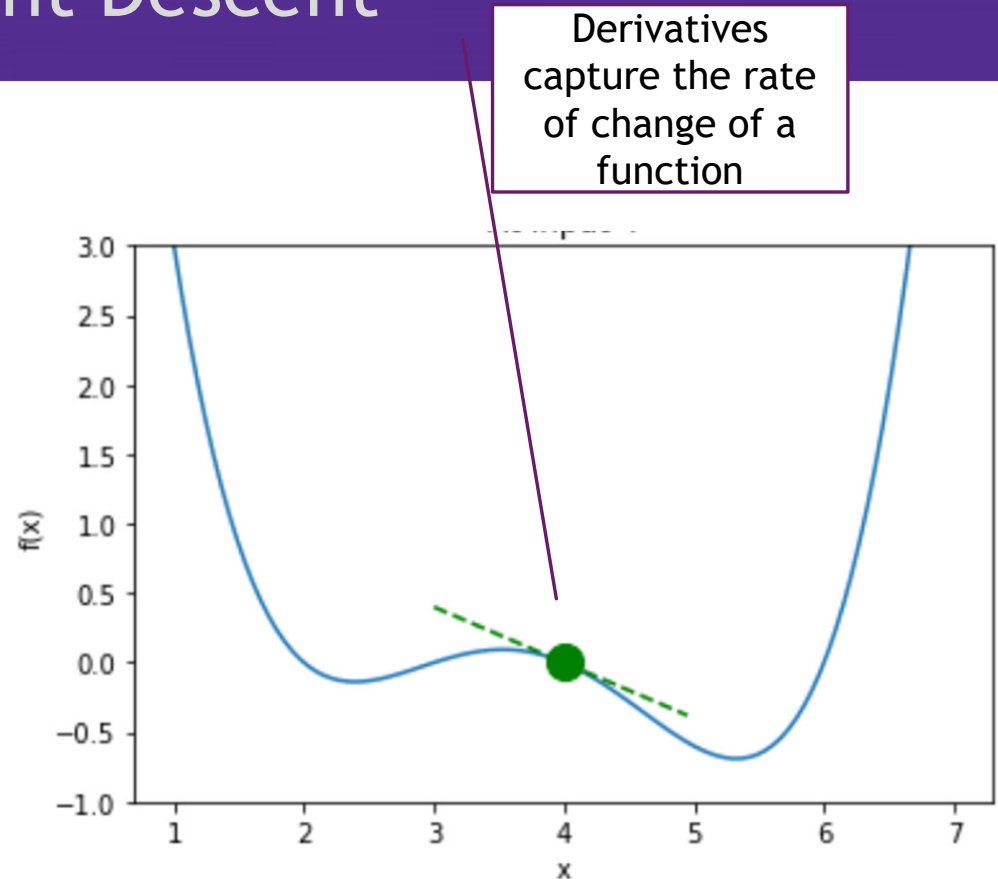
$$\log_b(a) = c \iff b^c = a$$

- If the independent variable and dependent variable have different scales, then we can apply logarithms to straighten out the data.



Gradient Descent

- ▶ By minimizing the average loss, we obtain parameters that fit the model to the data.
- ▶ We should not guess inputs and check outputs to minimize the function because that approach is inefficient and inaccurate.
- ▶ Instead we will just make one guess and use the derivative to update the guess. We call the approach **gradient descent**.

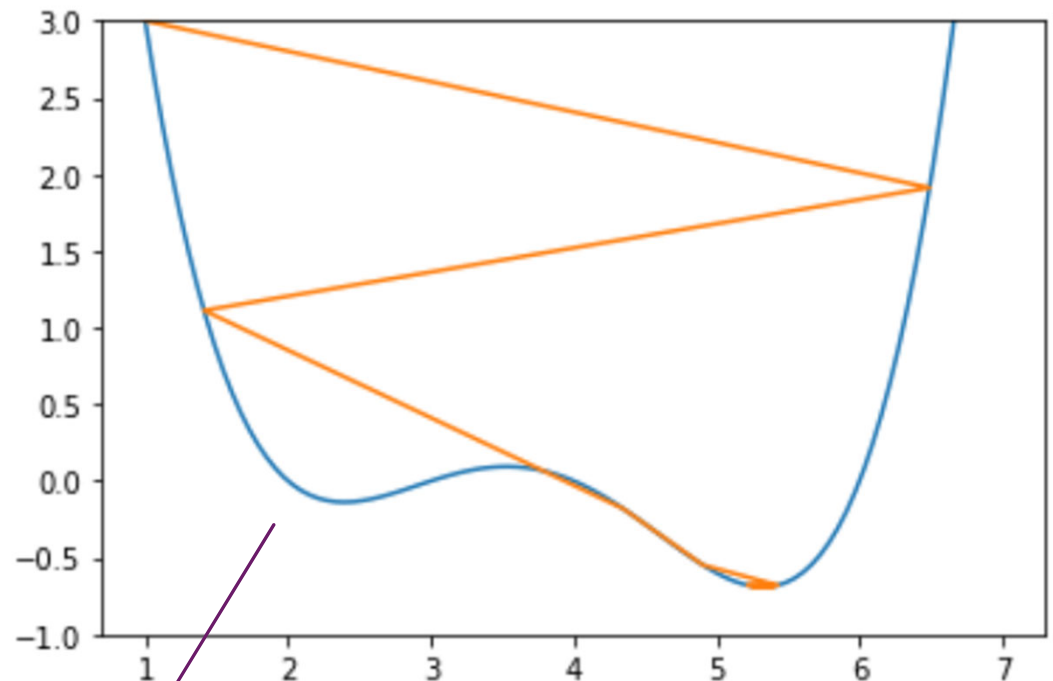


Gradient Descent

- ▶ Starting from an initial guess $x^{(0)}$ we update the guess with the formula

$$x^{(t+1)} = x^{(t)} - \alpha \frac{d}{dx} f(x)$$

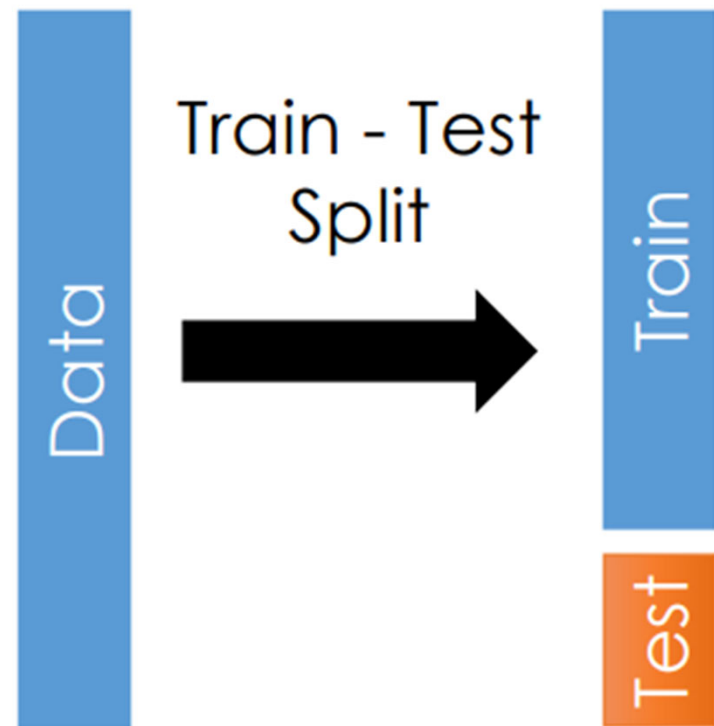
- ▶ Here α denotes the **learning rate**. If α is large, then guesses can change a lot between iterations. If α is small, then guesses can change a little between iterations



If the learning rate is too large then gradient descent might diverge from the minimum

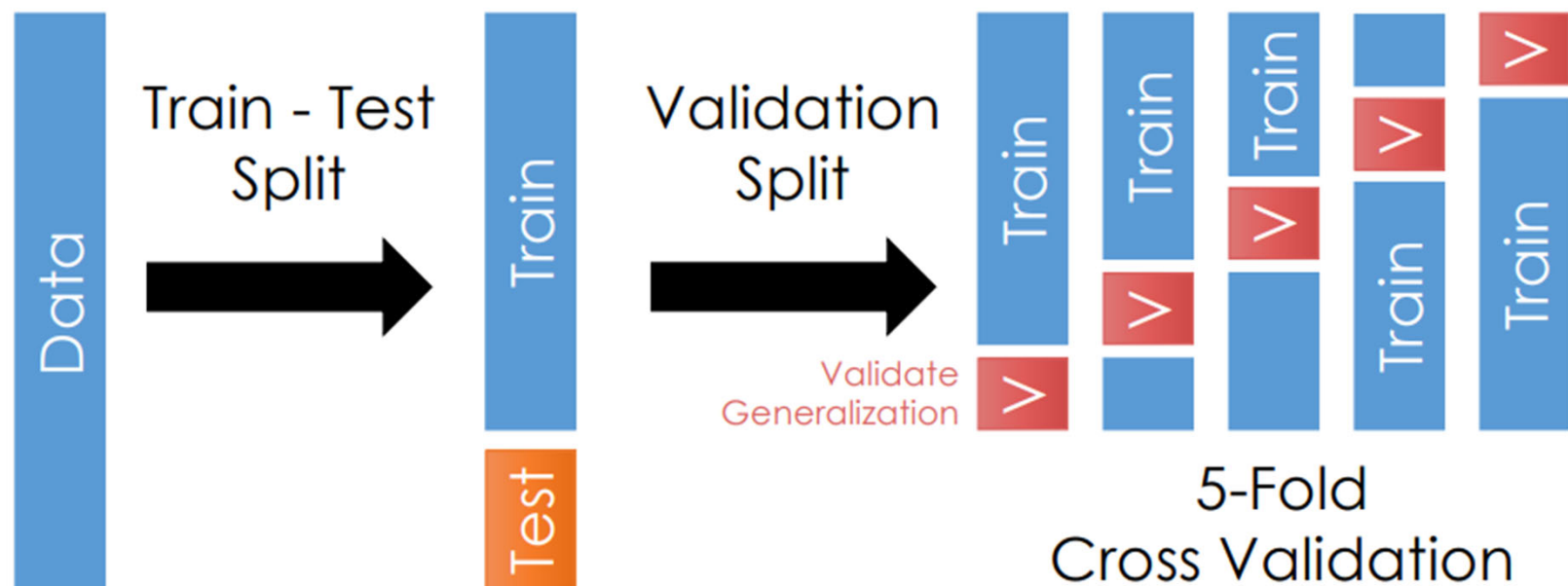
Testing Set

- ▶ Instead of studying one sample, we need to study two samples
 - ▶ training set
 - ▶ testing set
- ▶ We will fit the model to the data in the training set. We will check the accuracy of the predictions on the testing set.
- ▶ Usually we take 80% of the data for the training set and 20% of the data for testing set.



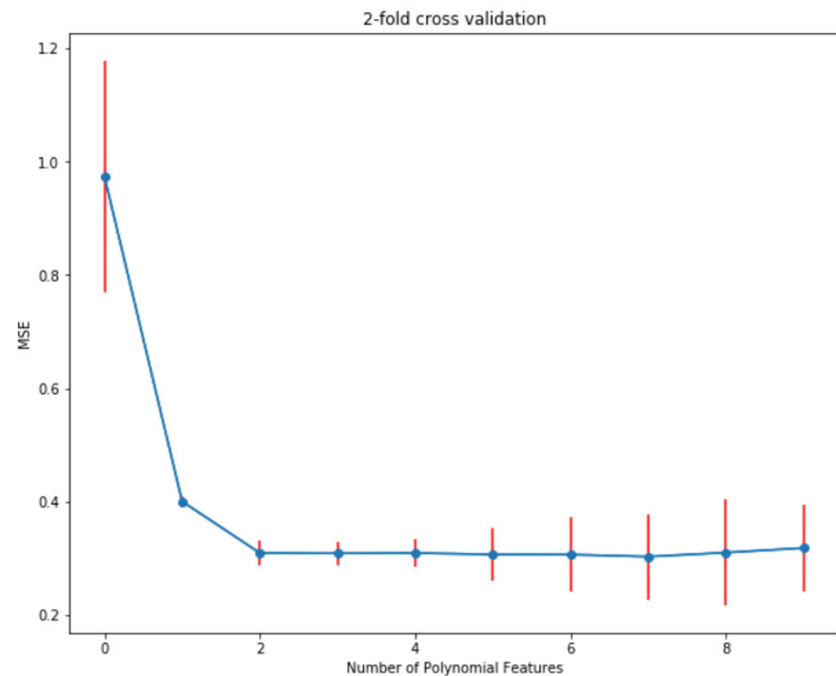
Validation Set

- We split one sample into two sample to generate the training set and testing set. Next we split the training set into k folds. Each fold has a training set and a **validation set**



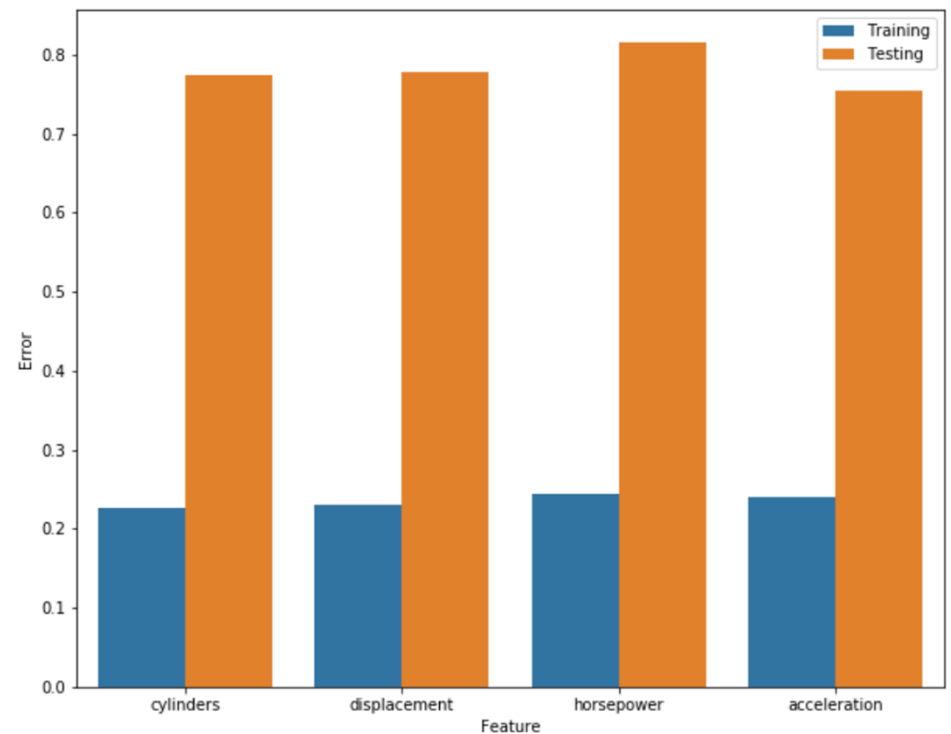
Cross Validation

- ▶ We want to choose models that are both **accurate** and **consistent**.
- ▶ With cross validation we measure the difference between predictions and observations on many datasets.
 - ▶ Small errors give us accuracy
 - ▶ Similarity between errors give us consistency
- ▶ We can visualize both the accuracy and consistency through a line chart with **error bars**



Feature Selection

- ▶ If we want to remove features to prevent against overfitting, then we could try to assess the effect of dropping combinations of features.
- ▶ In **backward feature selection** we
 - ▶ select a feature
 - ▶ drop it from the table
 - ▶ fit a model to the data
 - ▶ calculate average loss
- ▶ The feature that led to the smallest increase in loss should be excluded from predictions



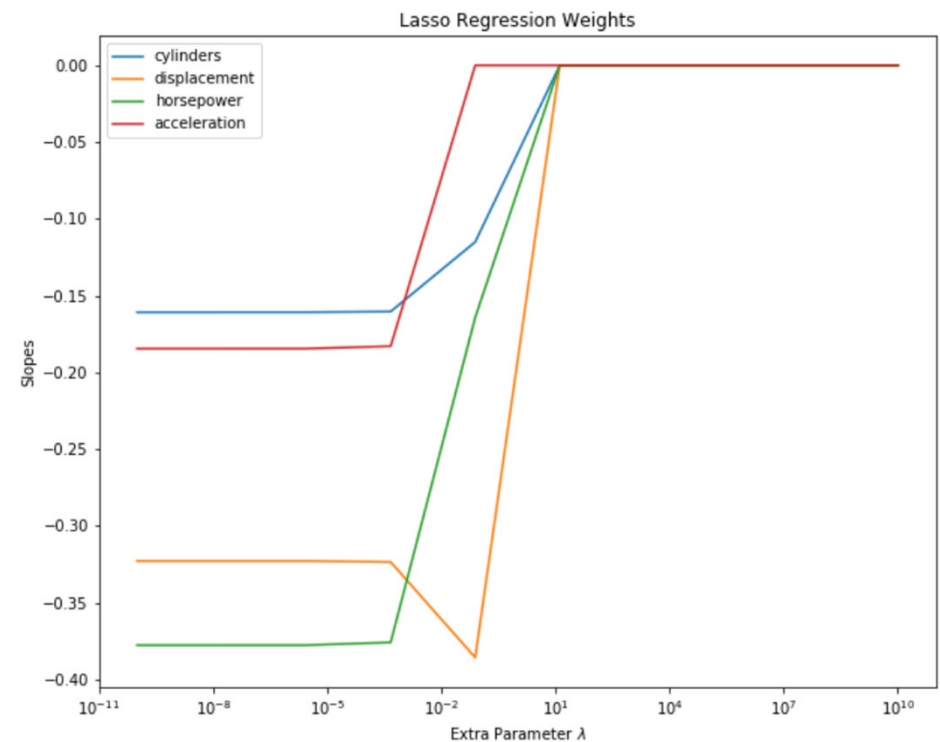
Lasso Regression

- We can replace the average loss for linear regression

$$\frac{1}{n} \sum_{i=1}^n (a + bx_i - y_i)^2$$

with regularized average loss for
lasso regression

$$\lambda (|a| + |b|) + \frac{1}{n} \sum_{i=1}^n (a + bx_i - y_i)^2$$



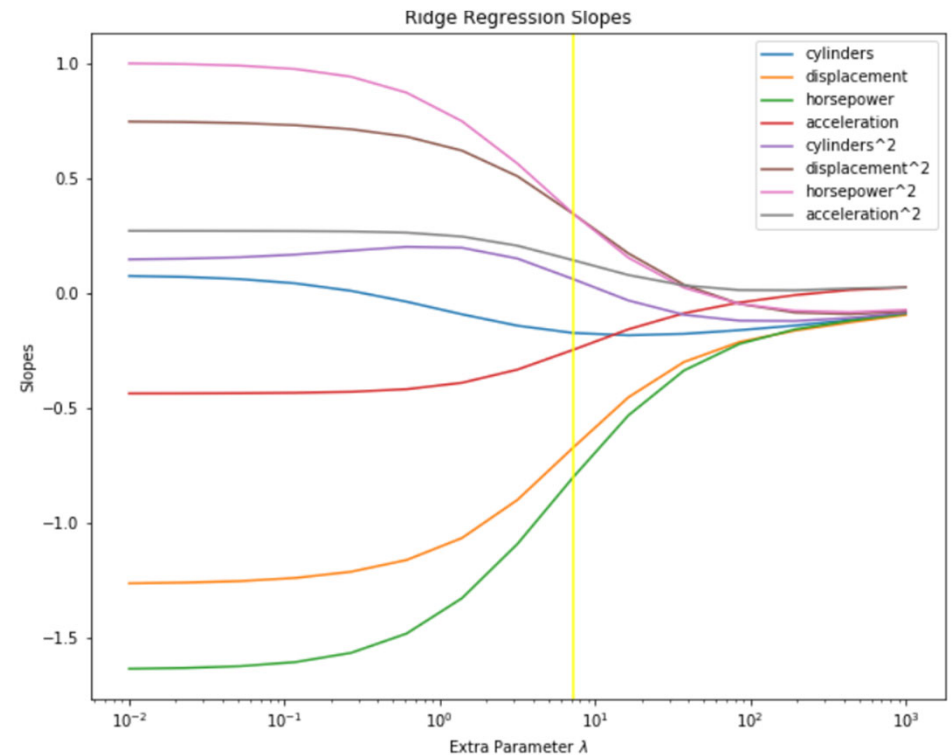
Ridge Regression

- We can replace the average loss from linear regression

$$\frac{1}{n} \sum_{i=1}^n (a + bx_i - y_i)^2$$

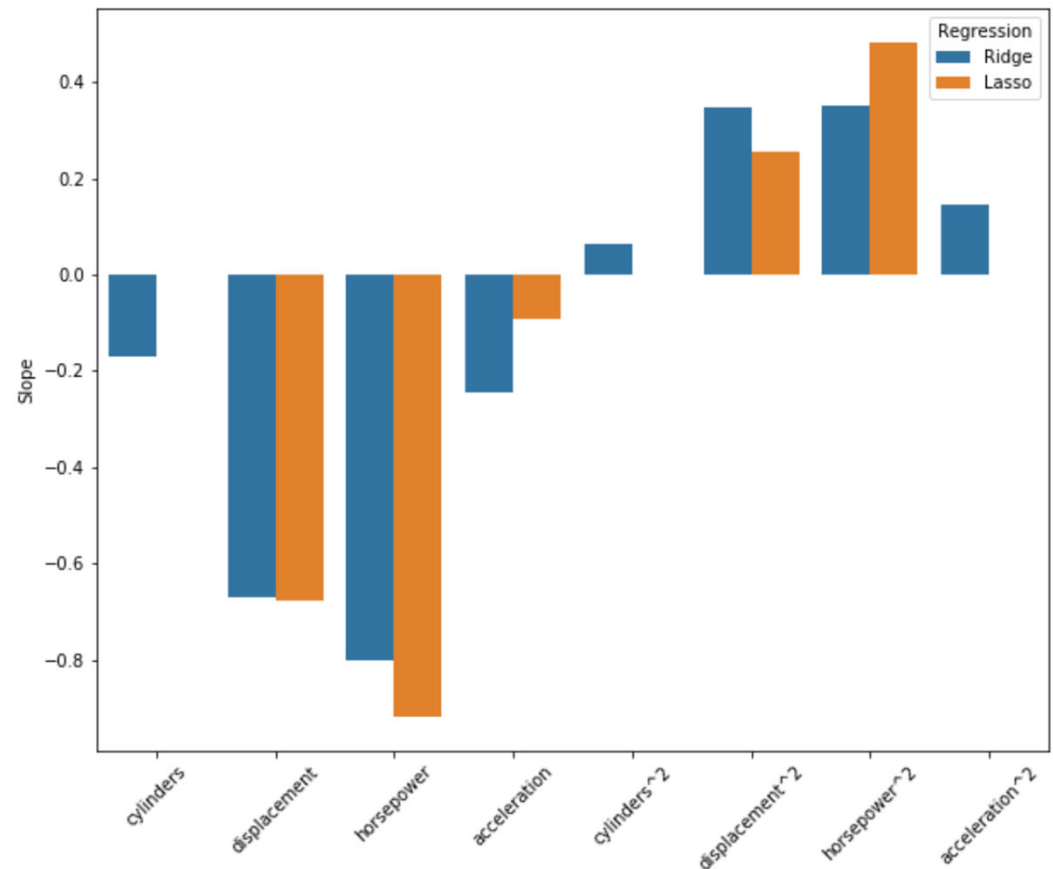
with regularized average loss for ridge regression

$$\lambda (a^2 + b^2) + \frac{1}{n} \sum_{i=1}^n (a + bx_i - y_i)^2$$



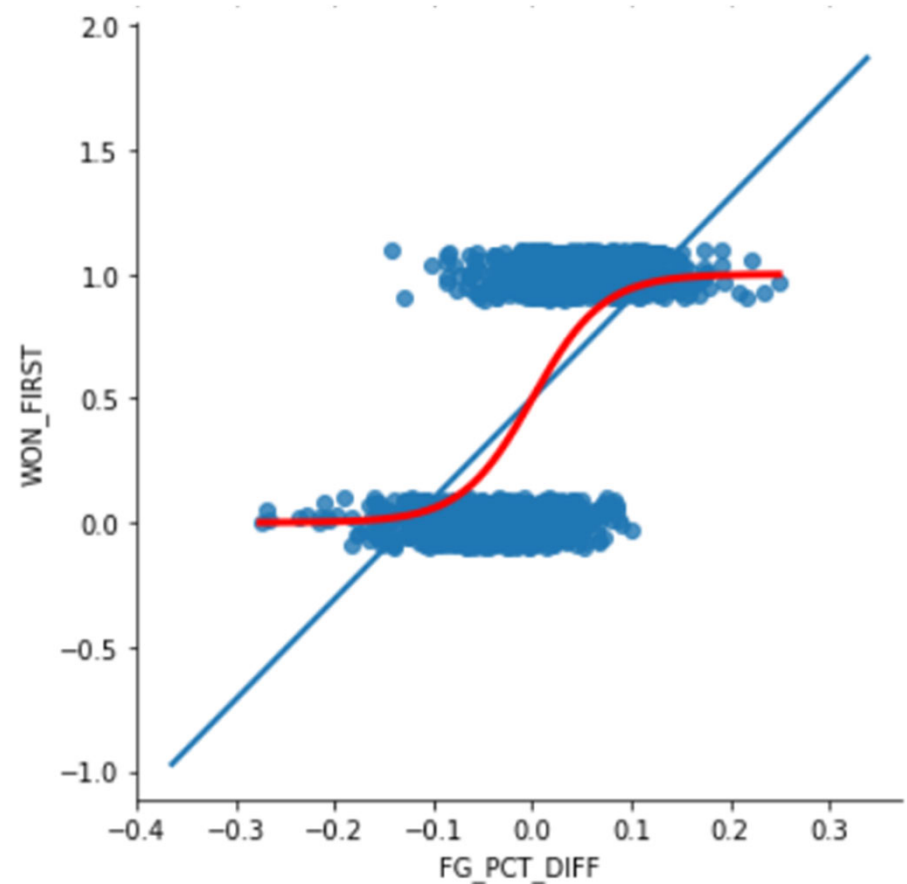
Regularization

- ▶ Lasso Regression tends to shrink parameters down to zero.
 - ▶ Helpful to eliminate features from the model
 - ▶ Erratically chooses between associated features
- ▶ Ridge Regression tends to shrink parameters close to zero.
 - ▶ Helpful to average out the values of the parameters among associated features
 - ▶ Cannot eliminate features from the model



Logistic Regression

- ▶ We use linear regression to predict a quantitative response variable.
- ▶ We use **logistic regression** to predict a qualitative response variable.
- ▶ Usually we encode the categories with numbers like 0 and 1

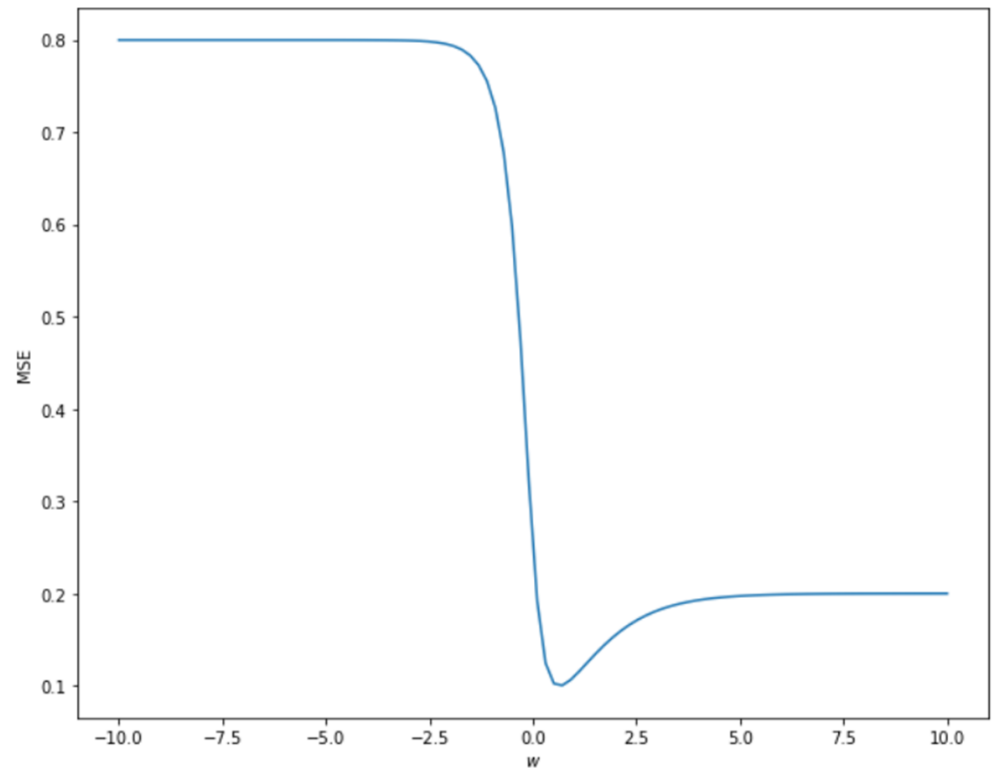


Logistic Loss

- Instead of the square loss we should take the **logistic loss**.

$$-y \log \hat{y} - (1 - y) \log(1 - \hat{y})$$

- Unlike the square loss, the logistic loss does not generate flat regions that prevent gradient descent from finding the minimum.



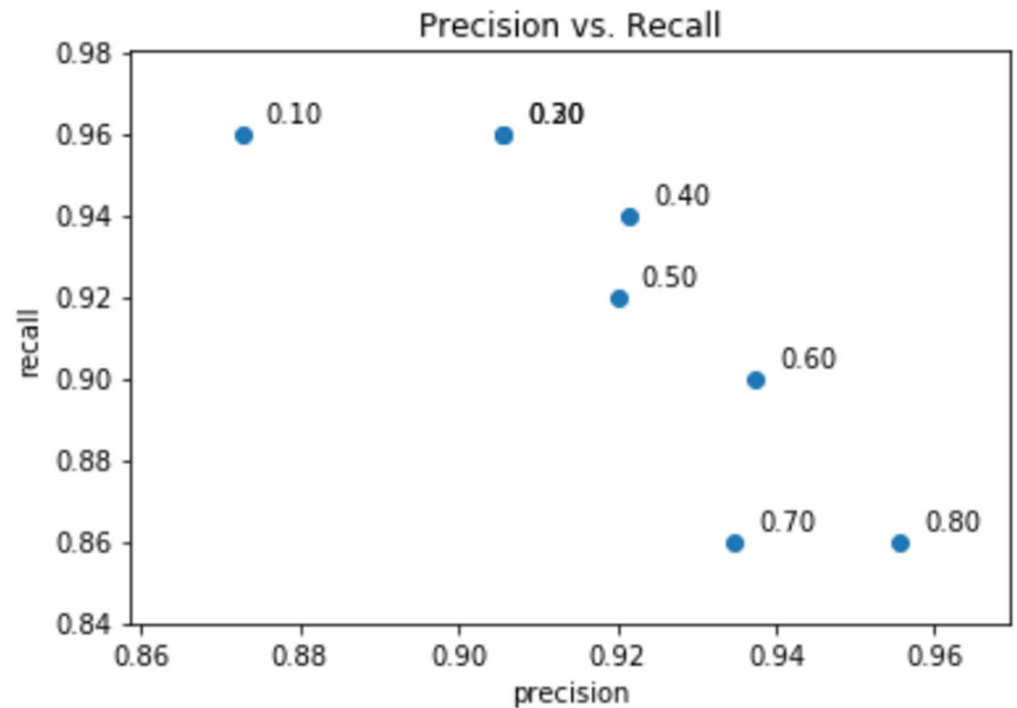
Confusion Matrix

- ▶ The observation take the value 1 or 0. The predictions take the value 1 or 0. So we have four possibilities
 - ▶ True Positive
 - ▶ False Positive
 - ▶ False Negative
 - ▶ True Negative
- ▶ We can visualize the number of each possibility for a dataset with a **confusion matrix**

		Observed	
Predicted	1	46	4
	0	4	83
		1	0

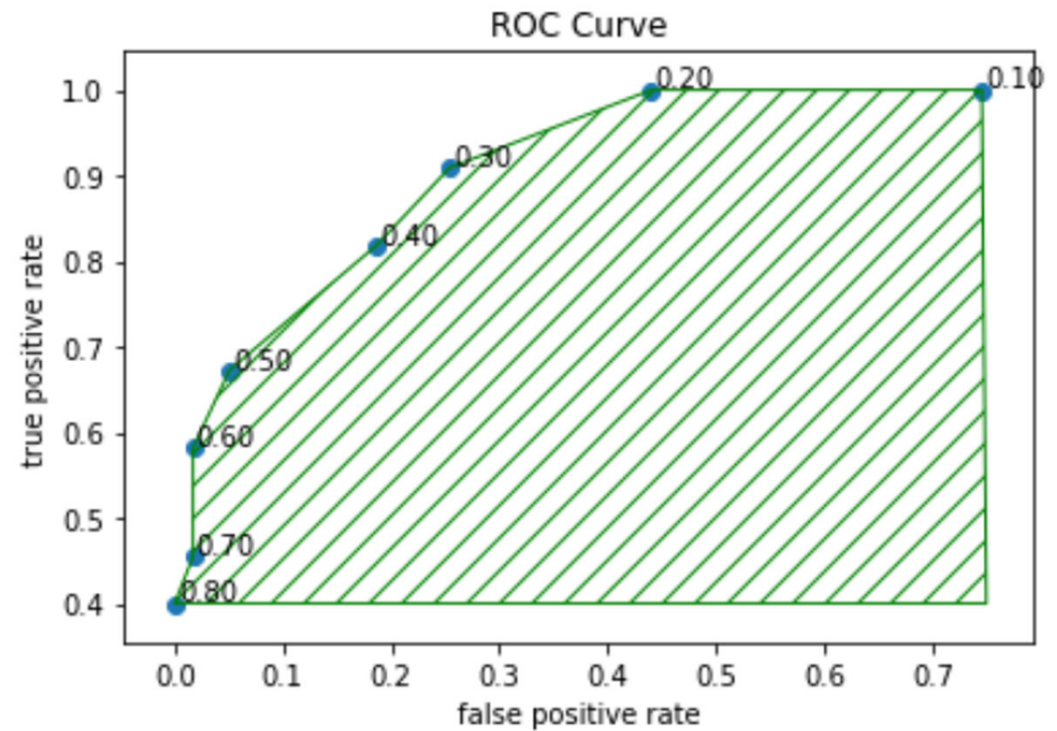
Precision Recall Curve

- ▶ Accuracy might not capture the differences between observations and prediction with an **imbalance** between categories
 - ▶ Precision penalizes **false positives**
 - ▶ Recall penalizes **false negative**
- ▶ We can visualize the trade-off between recall and precision through a **precision-recall curve**



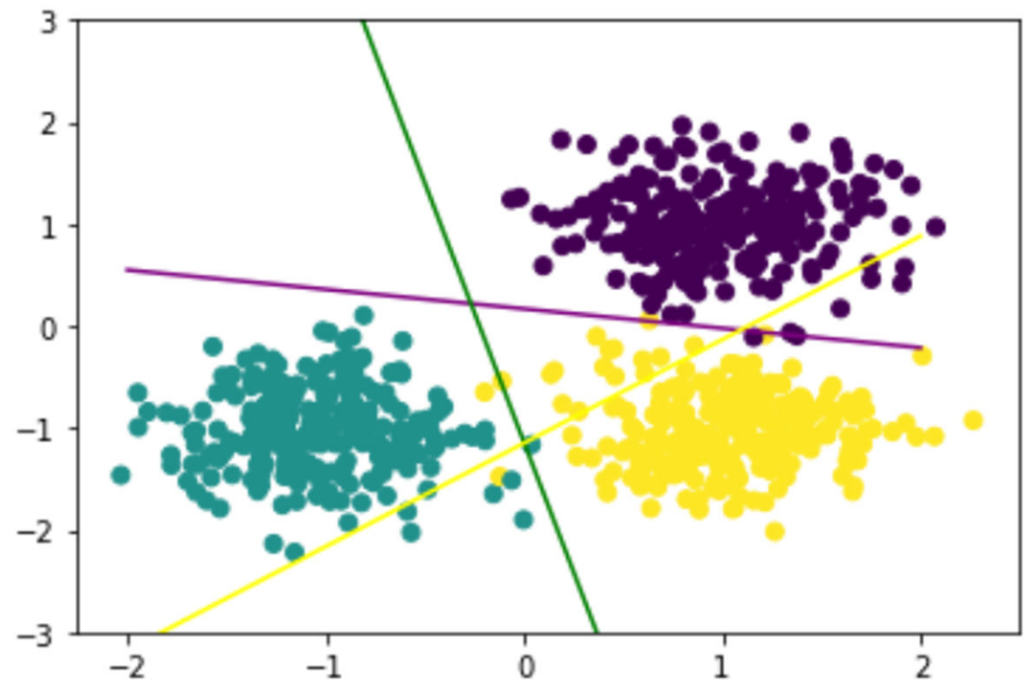
ROC Curve

- ▶ A **ROC** curve plots the true positive rate and the false positive rate
- ▶ The acronym ROC stands for Receiver Operating Characteristic.
- ▶ We can summarize the ROC curve with the area under the curve. We abbreviate the area under the curve as **AUC**.



Multiple Categories

- ▶ If we have three or more categories, then we can split the classification problem into multiple problems with two categories.
- ▶ Each problem try to classify one category versus the other categories. We call the approach **One-versus-Rest**.



Nearest Neighbors

- ▶ We determine the category of the unlabeled records from the categories of the nearest labeled records.
- ▶ If we predict categories for many unlabeled records then we can determine the decision boundary

