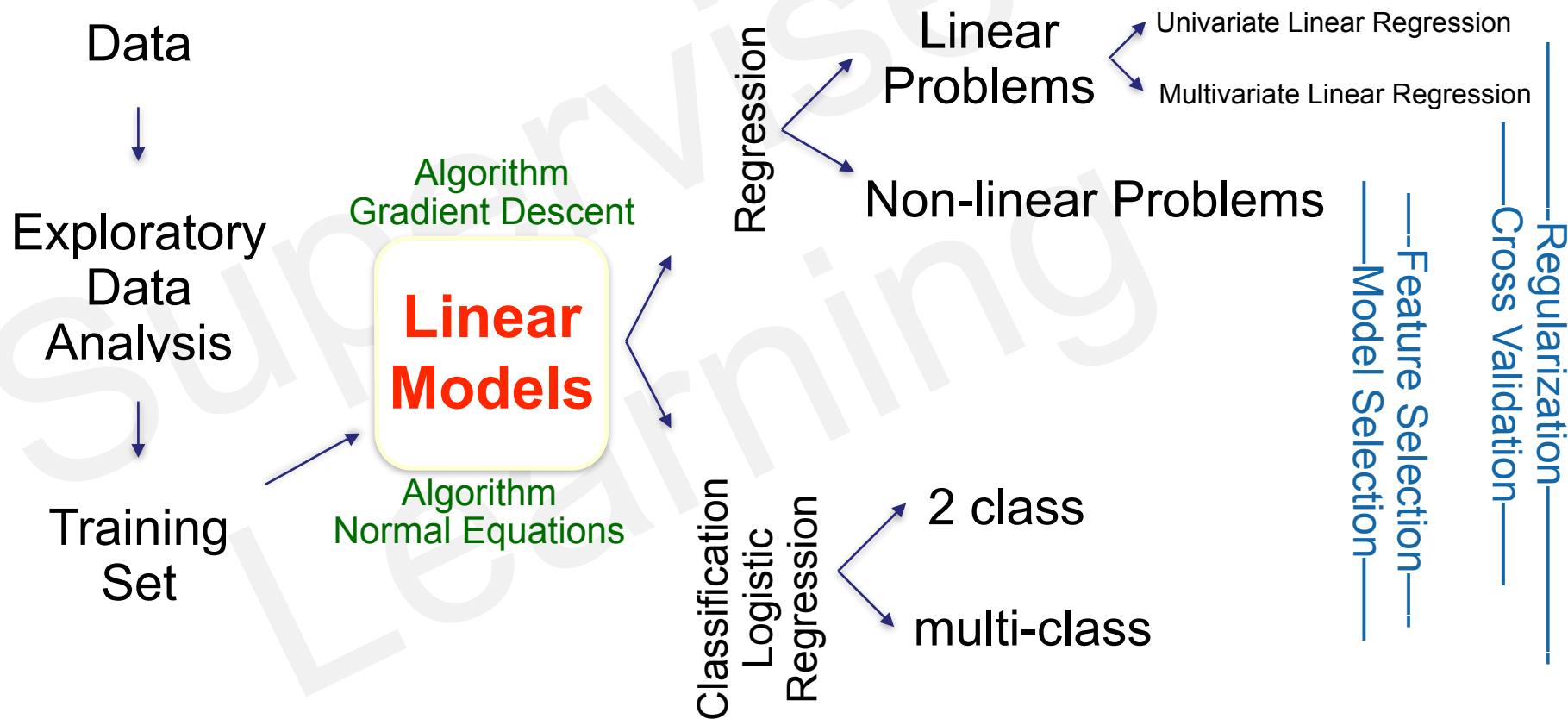


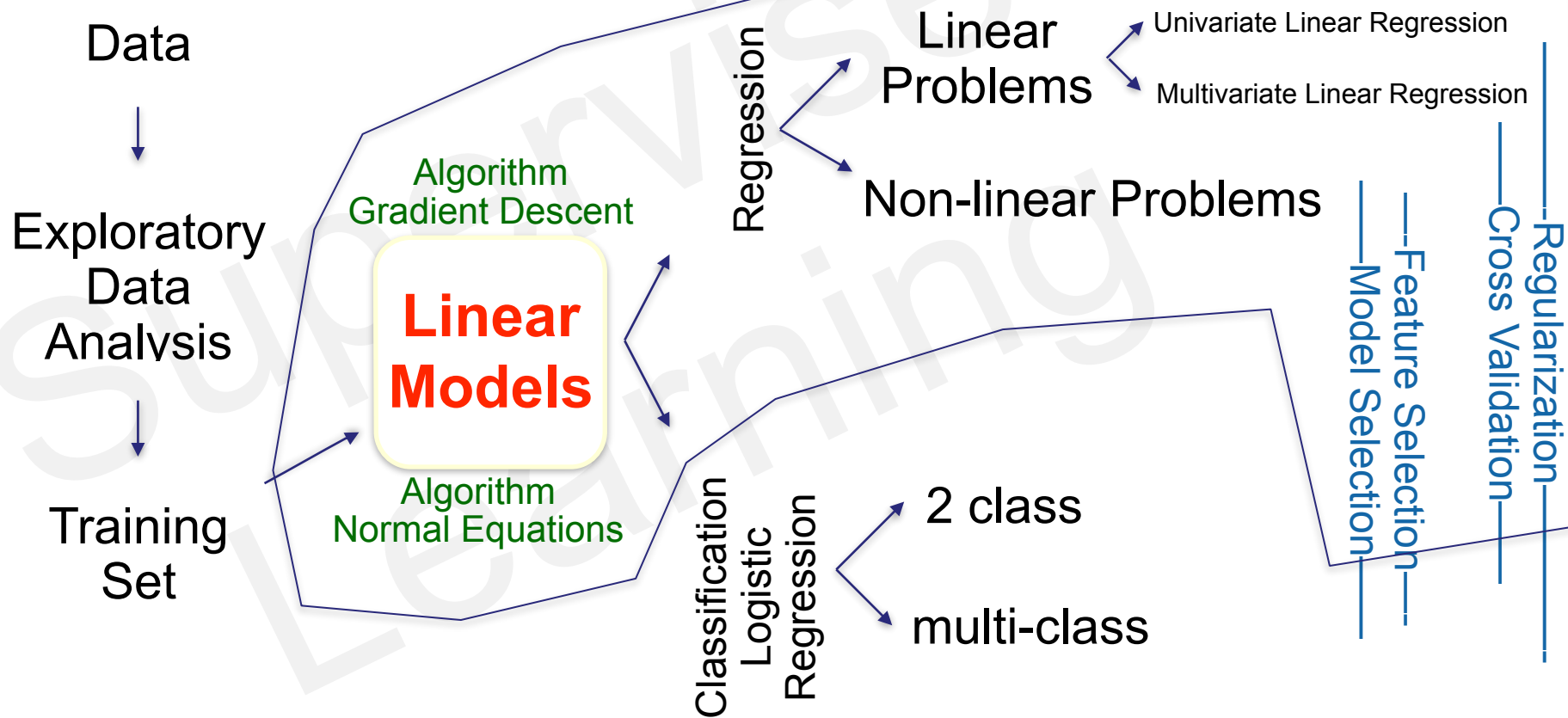
INTRO TO DATA SCIENCE

LECTURE 7: FEATURE SELECTION, MODEL SELECTION, REGULARIZATION

WHERE ARE WE ON THE DATA SCIENCE ROAD-MAP?



WHERE ARE WE ON THE DATA SCIENCE ROAD-MAP?



KEY CONCEPTS - FEATURE SELECTION

- Choosing the inputs to your model
- Depending on the number of features you have you might decide to adopt a 'brute force' approach
- e.g. africa soil inputs

KEY CONCEPTS - MODEL SELECTION & FEATURE SELECTION

- Both Feature Selection and Model Selection need to be optimized
- The mechanism by which you choose which features to use and how complex the model should be is a matter of judgement. If the search space is large then brute force is not really an option
- Validation is the mechanism by which you optimize over a set of features and models

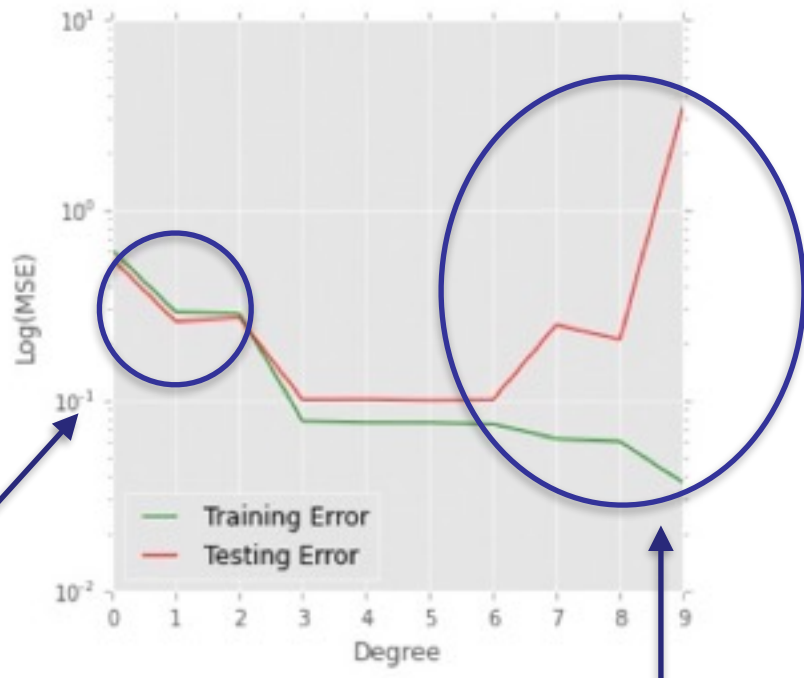
KEY CONCEPTS - REGULARIZATION

Linear models are prone to:

- under-fitting (bias), and
- over-fitting (variance)

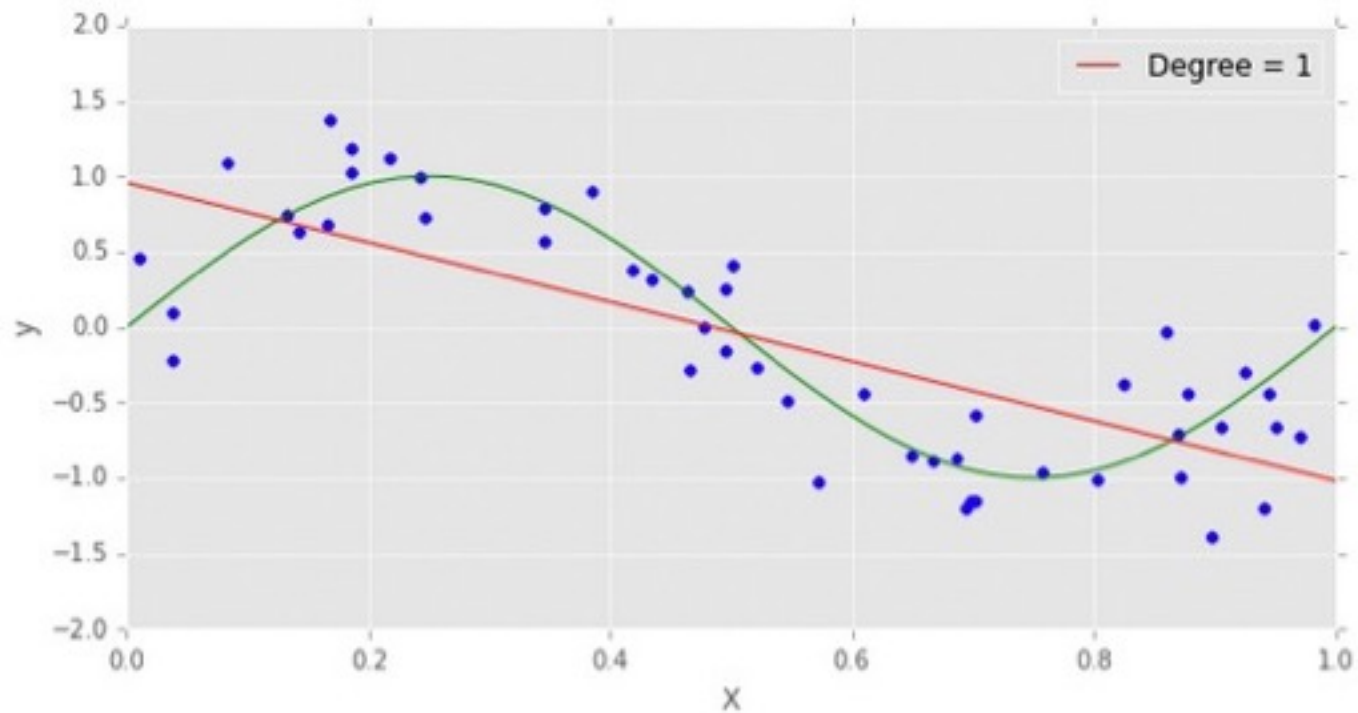
What does this actually look like?

Under-fit
High Bias

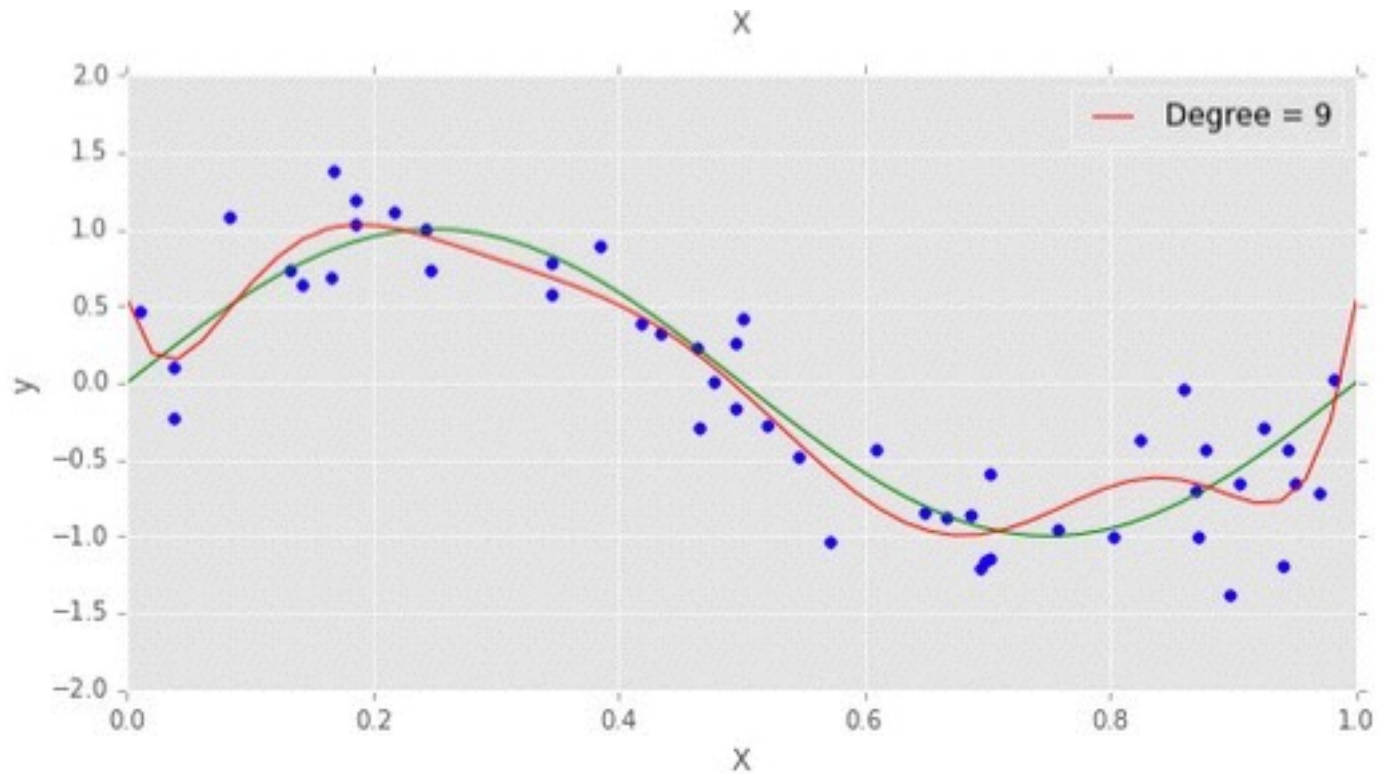


Over-fit
High Variance

KEY CONCEPTS - HIGH BIAS - UNDER-FITTING



KEY CONCEPTS - HIGH VARIANCE - OVER-FITTING



KEY CONCEPTS - OVER-FITTING

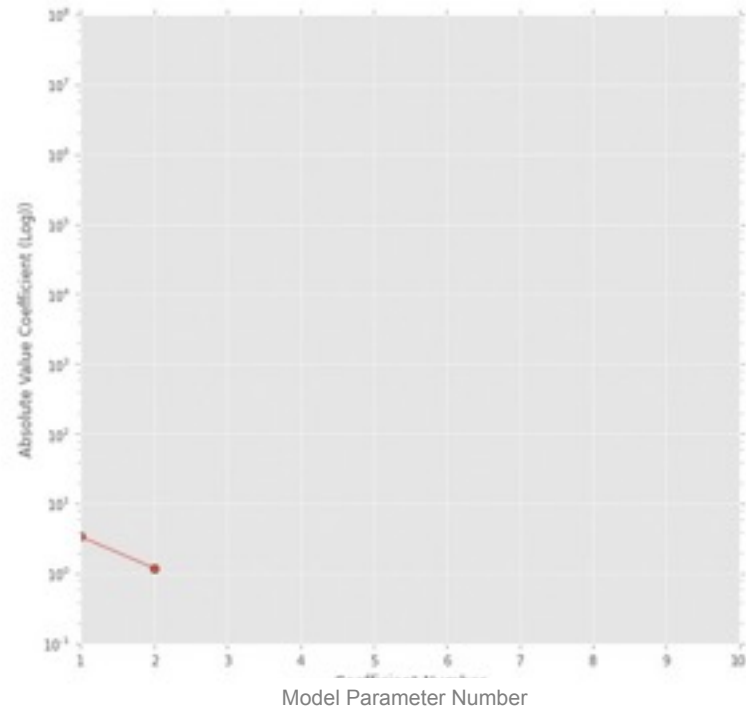
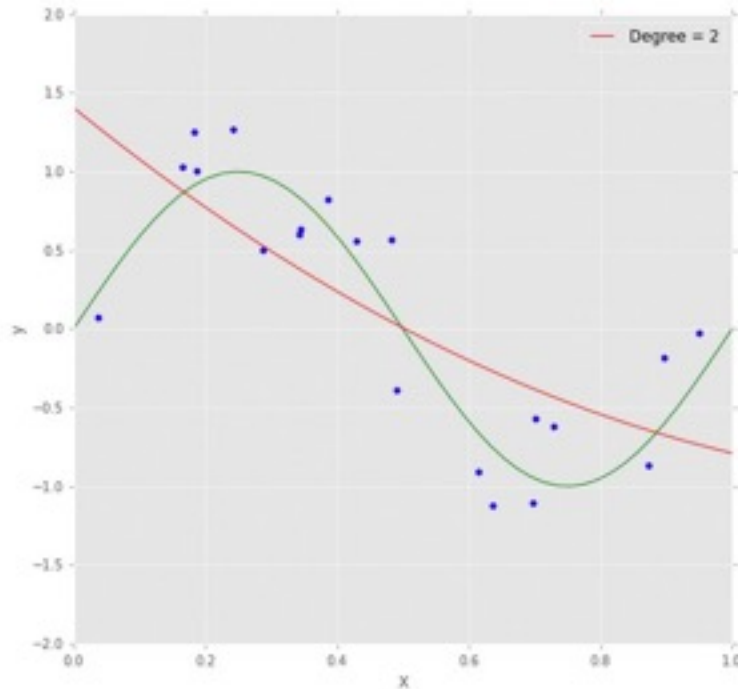
- Too many features and/or too complex a model can cause over-fitting
- While it may be tempting to reduce the complexity of a model and/or reduce the number of features over-fitting can be controlled by regularization
- In general the complexity of your features (number and kind) should be determined by the complexity of the problem you are trying to solve AND NOT by a desire to get the model to fit the data

KEY CONCEPTS - REGULARIZATION

- Once features have been chosen - number and type, then fitting the model is controlled by regularization
- Linear models should always be regularized
- To see what regularization achieves let's examine the size of our model parameters (θ) as over-fitting occurs
- Understanding this will suggest a solution to the problem

KEY CONCEPTS - UNDER & OVER-FITTING

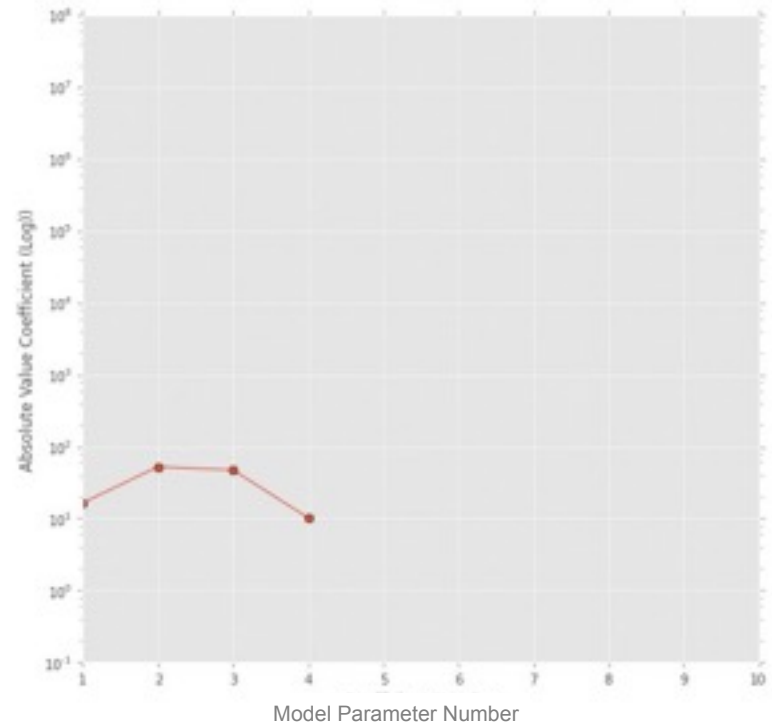
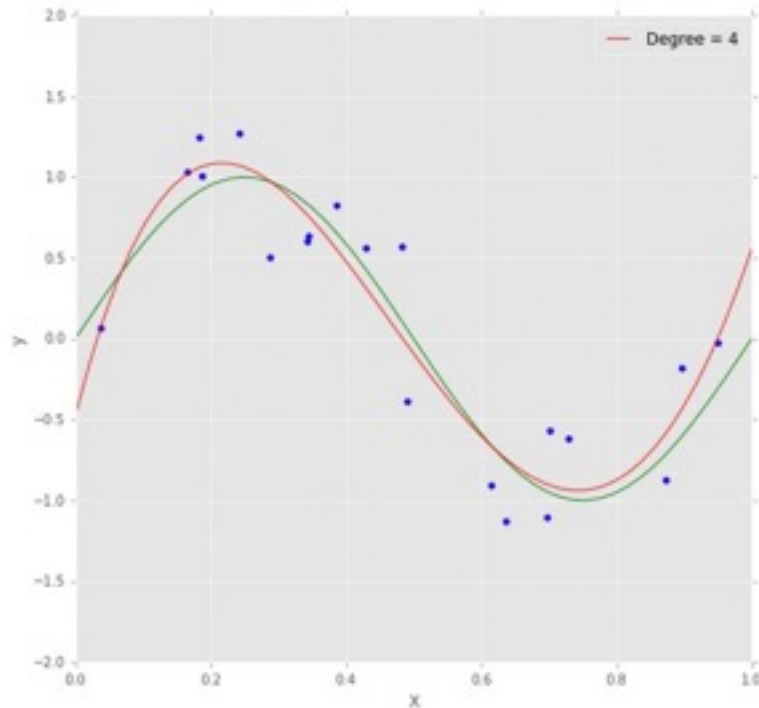
Here we have a simple model that is under-fitting the training data (blue points)
The magnitude of the model parameters is between 1 and 10



KEY CONCEPTS - UNDER & OVER-FITTING

A more complex model that fits well.

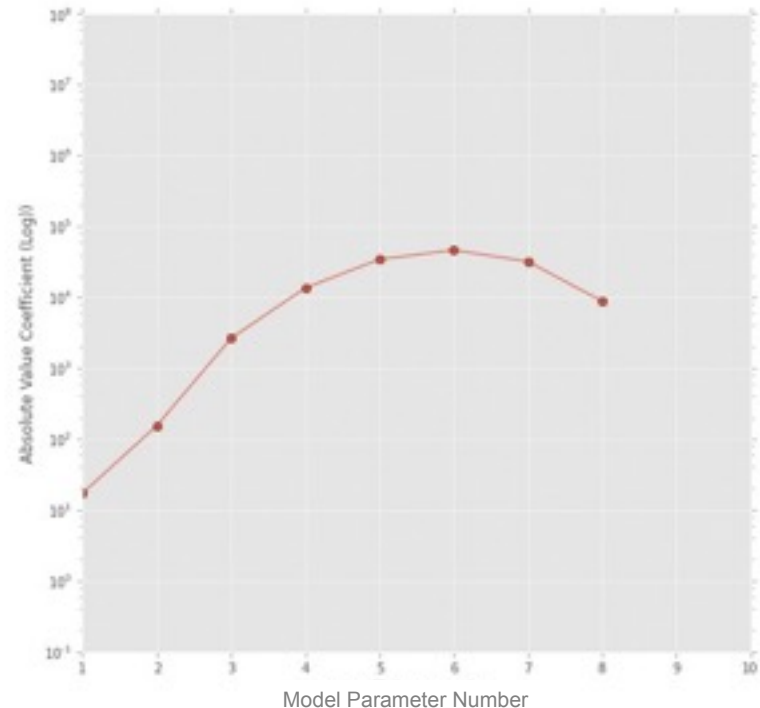
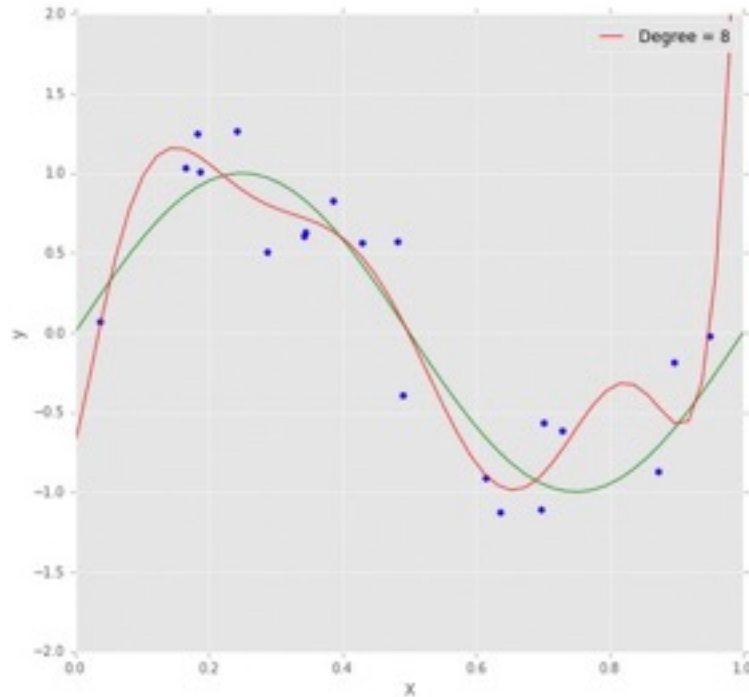
The magnitude of the model parameters is between 10 and 100



KEY CONCEPTS - UNDER & OVER-FITTING

This model is over-fitting the training data

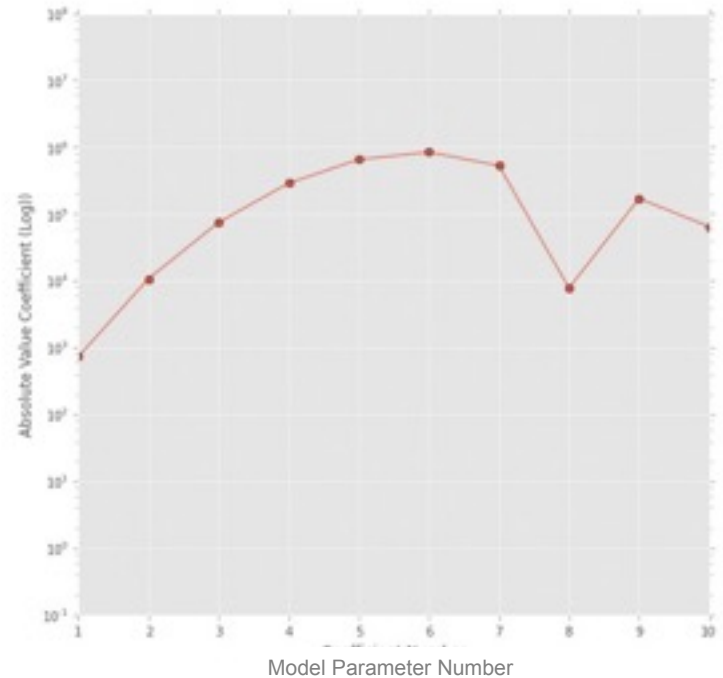
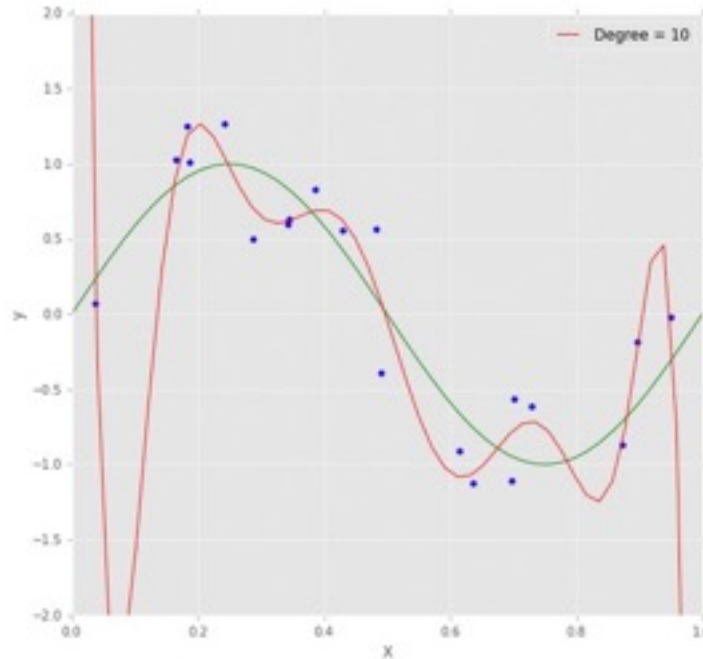
The magnitude of the model parameters is between 10 and 100000!



KEY CONCEPTS - UNDER & OVER-FITTING

Severe over-fitting

The magnitude of the model parameters is between 100 and 1000000!!



KEY CONCEPTS - THE NEED FOR REGULARIZATION

- Over-fitting is associated with larger model parameter magnitudes
- Therefore, one solution might be to penalize large parameters while fitting the model
- This is our original cost function - minimizing sum of squares

$$J(\theta) = \frac{1}{2m} \sum_{i=1}^m (y_i - \hat{y}_i)^2$$

KEY CONCEPTS - THE NEED FOR REGULARIZATION

- What would happen to the magnitudes of the model parameters if we modified the cost function like this:

$$J(\theta) = \frac{1.0}{2m} \left[\sum_{i=1}^m (y_i - \hat{y}_i)^2 + 1000 * \theta_3 + 1000 * \theta_4 \right]$$

- Not only would this minimize the sum of squared errors but the two model parameters would also be constrained. Large model parameter values would be penalized

KEY CONCEPTS - THE NEED FOR REGULARIZATION


- In general, therefore, we modify the cost function to be:

$$J(\theta) = \frac{1.0}{2m} \left[\sum_{i=1}^m (y_i - \hat{y}_i)^2 + \lambda \sum_j^N \theta_j \right]$$

- λ is called the regularization parameter. The bigger λ is the more the magnitudes of θ will be penalized

KEY CONCEPTS - THE NEED FOR REGULARIZATION

- The story doesn't end there, however. The exact mathematical form of the regularization parameters can be altered to penalize the model parameters in different ways.


$$J(\theta) = \frac{1.0}{2m} \left[\sum_{i=1}^m (y_i - \hat{y}_i)^2 + \lambda \sum_j^N \theta_j \right]$$

KEY CONCEPTS - RIDGE REGRESSION, LASSO REGRESSION

- Ridge Regression and Lasso Regression are two forms of regression where the regularization formulae differ
- Both accept regularizing parameters
- But the resulting models returned by these algorithms are significantly different

KEY CONCEPTS - RIDGE REGRESSION, LASSO REGRESSION

- Ridge (Tikhonov regularization) = L2-norm = Euclidean norm of the sum of the parameters, θ , of the model = $\lambda ||\theta||^2$
 - As the penalty is increased (λ) ALL parameters shrink, while still remaining non-zero
- Lasso (Least Absolute Shrinkage and Selection Operator) = L1-norm = Least Absolute Deviation = $\lambda ||\theta||$
 - As the penalty is increased MORE of the parameters will shrink to zero
 - This can discard features

KEY CONCEPTS - ELASTICNET

- Elastic Net is a linear combination of the Ridge and Lasso regularizers

http://scikit-learn.org/stable/modules/generated/sklearn.linear_model.Ridge.html

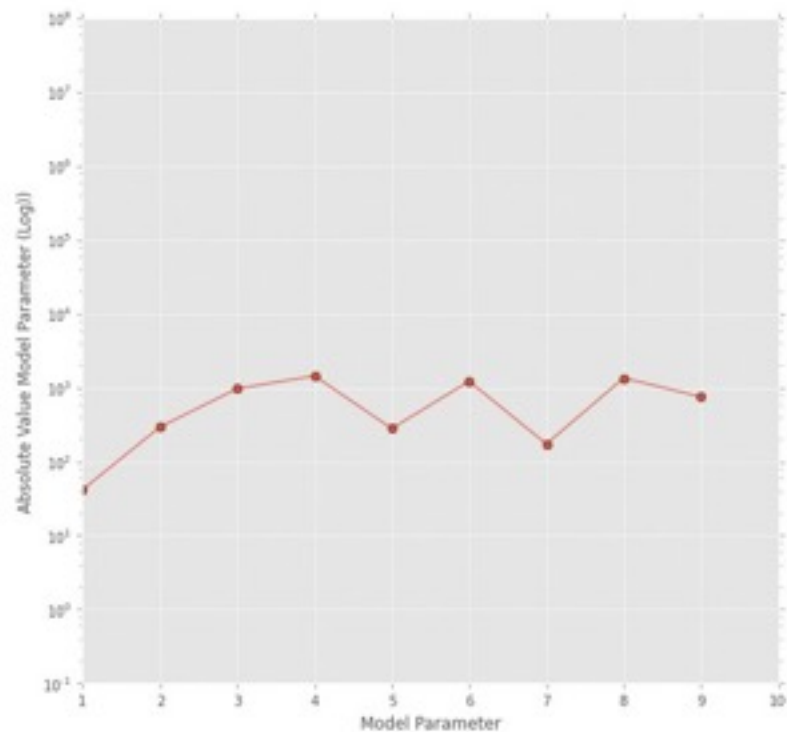
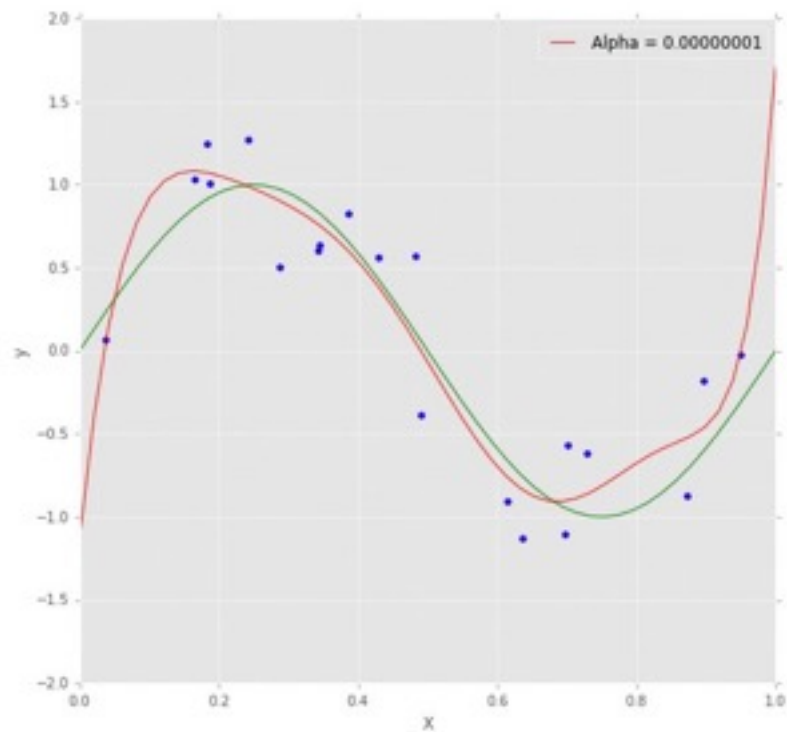
http://scikit-learn.org/0.11/modules/generated/sklearn.linear_model.Lasso.html

http://scikit-learn.org/stable/modules/generated/sklearn.linear_model.ElasticNet.html

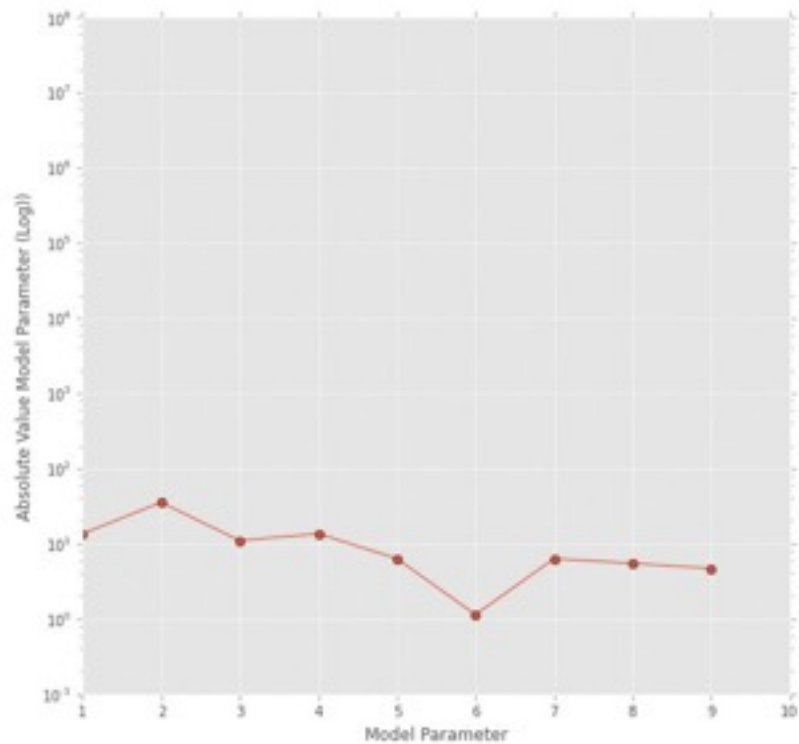
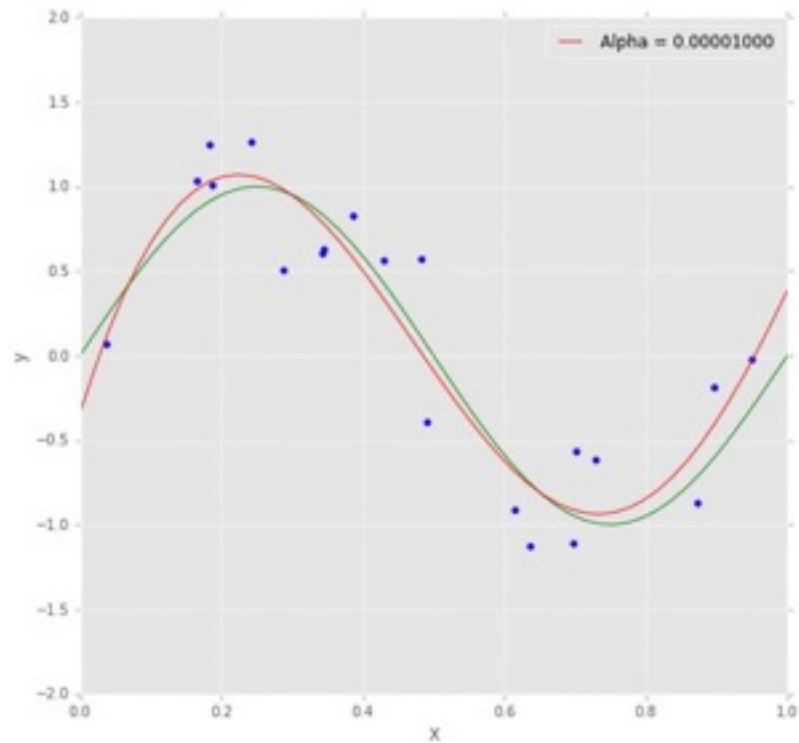
KEY CONCEPTS - RIDGE REGRESSION, LASSO REGRESSION

- Sklearn implementation - alpha is the regularizer
- Ridge, Lasso and ElasticNet internally use gradient descent
 - You will notice a 'maximum iterations' argument
- In short the algorithms take care of the method and mechanism of optimization for you

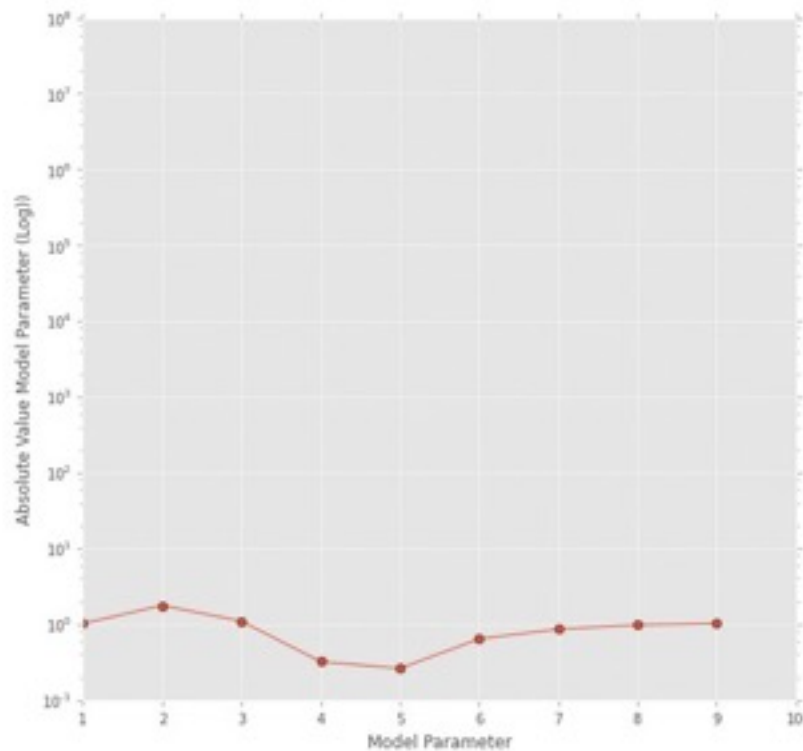
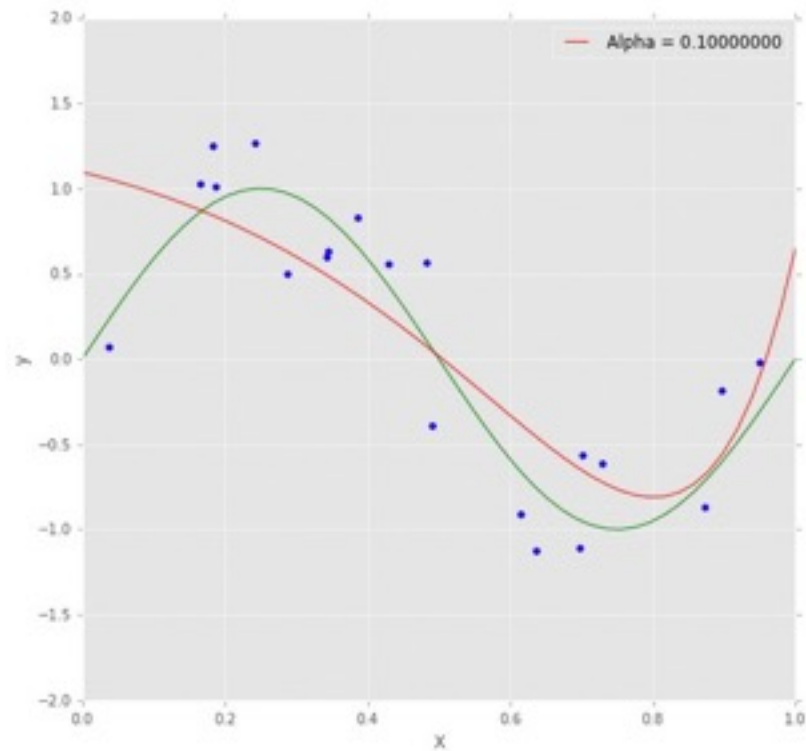
KEY CONCEPTS - RIDGE REGRESSION



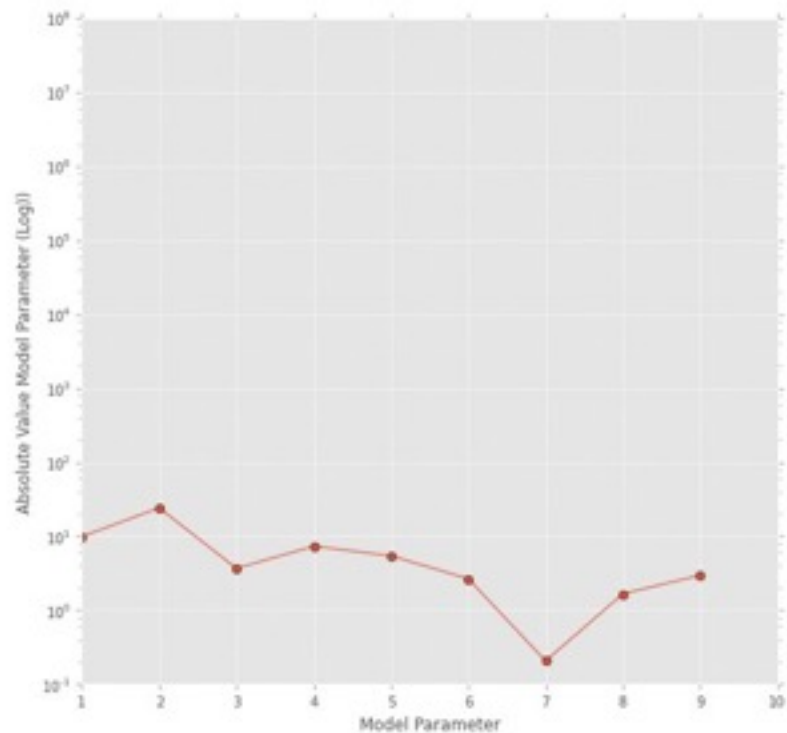
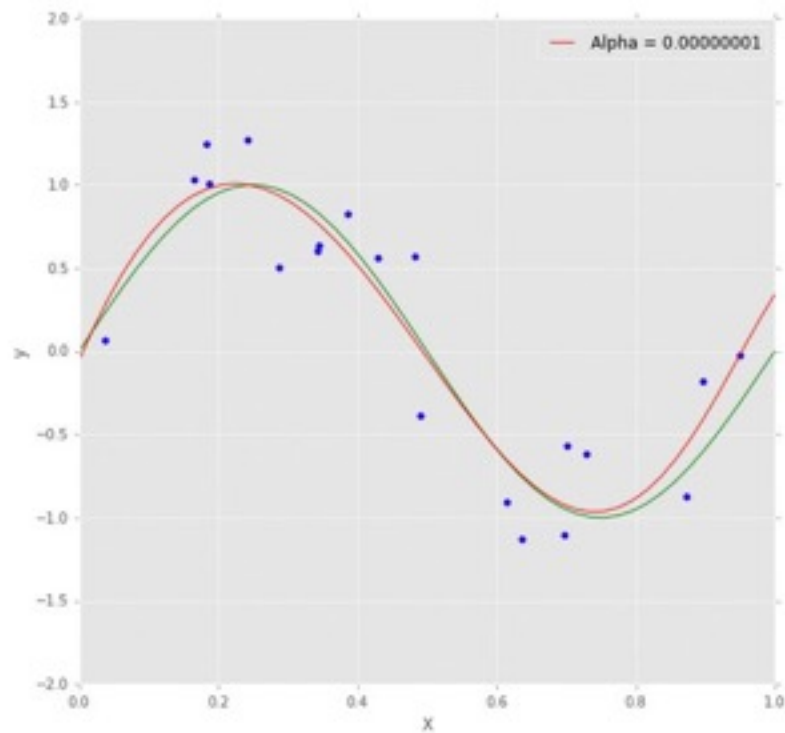
KEY CONCEPTS - RIDGE REGRESSION



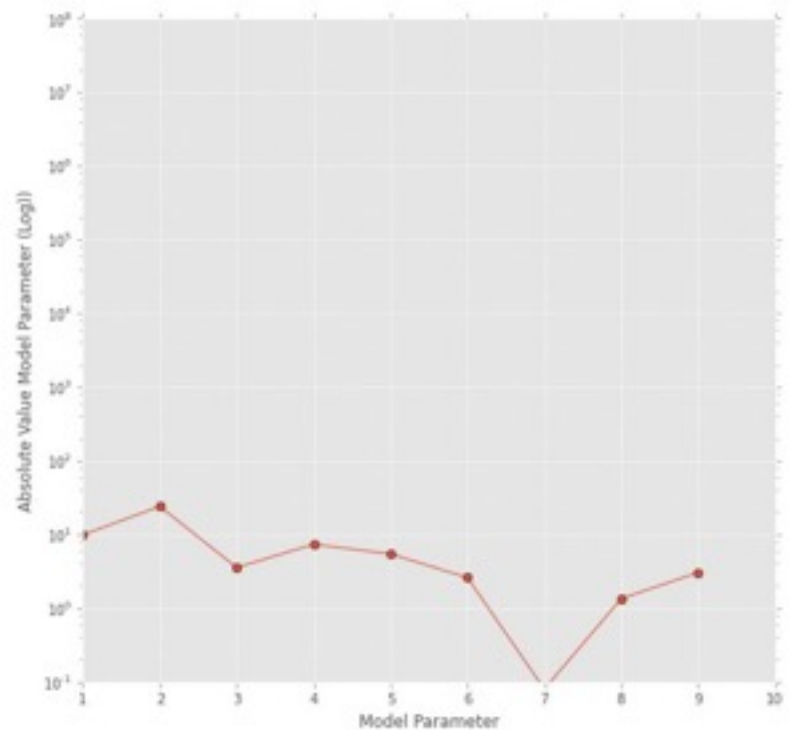
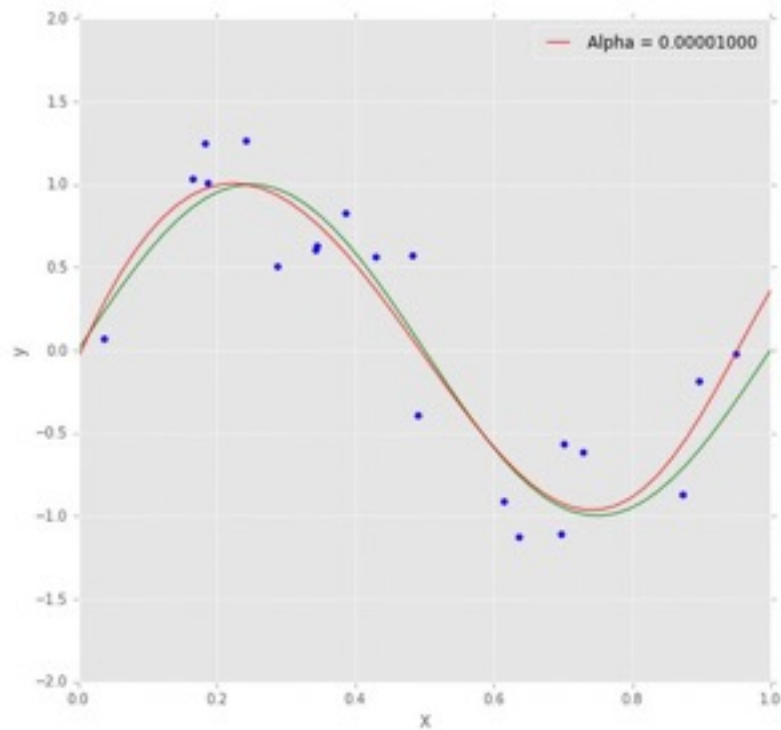
KEY CONCEPTS - RIDGE REGRESSION



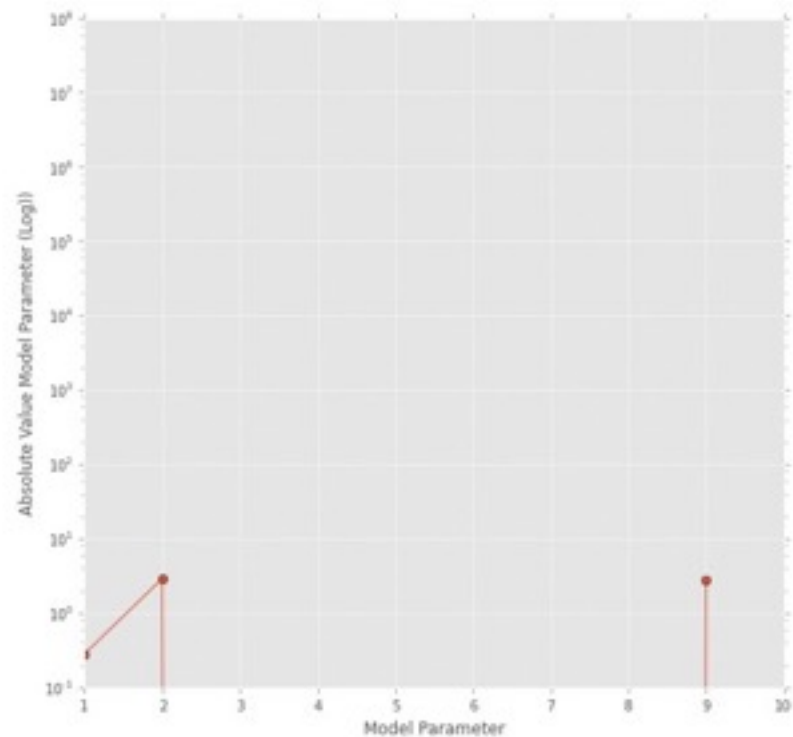
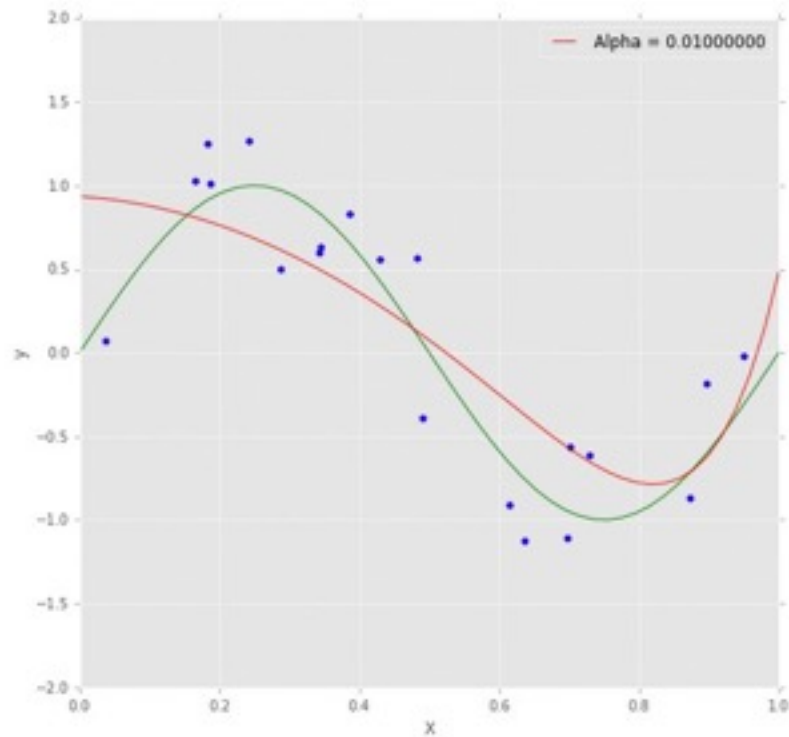
KEY CONCEPTS - LASSO REGRESSION



KEY CONCEPTS - LASSO REGRESSION



KEY CONCEPTS - LASSO REGRESSION



KEY CONCEPTS - REGULARIZATION IN SUMMARY

- The process of preventing over-fitting is known as regularization
- What are some ways of regularizing a model:
 1. Can use the complexity of a model as a regularizer
 - i. keep the model simple
 - ii. bad idea (?)
 - iii. the complexity of a model should match the complexity of the task you are trying to model
 2. Within the models themselves we can add a regularizing 'term'
 - i. instead of our cost function just minimizing MSE we use the cost function to “smooth between the points”

KEY CONCEPTS - IMPROVING THE PERFORMANCE OF A MACHINE LEARNING ALGORITHM

1. Get more training examples *
2. Try a smaller set of features
3. Try additional features
4. Try different features
5. Try different regularization parameters

* Be careful not to go down a blind avenue. Collecting more data, may, for example, cost a lot of money and cost a lot of time, and may yield not net improvement in the model!

KEY CONCEPTS - DIAGNOSTICS FOR A MACHINE LEARNING ALGORITHM

1. Both bias (under-fitting) and variance (over-fitting) result in poor generalization
2. Make sure you split your data
 - use the datasets correctly!
 - 70/30 or 60/20/20 (Equal size is ideal)
 - if data is limited use S-fold cross validation
3. Monitor a metric that will indicate good generalization performance = mse on the validation set

KEY CONCEPTS - DIAGNOSTICS FOR A MACHINE LEARNING ALGORITHM

4. Choose the model with the lowest mse on the validation set
5. If you have a test set then use it to report the results of your best model

KEY CONCEPTS - REGULARIZATION, BIAS AND VARIANCE

1. Plot the degree of polynomial vs training AND validation error
 - High bias = high training and validation error
 - High variance = low training/high validation error
2. Use regularization to prevent over-fitting
3. Increase the regularization parameter by a factor of 2

1. Plot the regularization parameters vs training error AND validation error

- High bias = large regularization parameter, high training AND high validation error
- High Variance = low regularization parameter, low training/ high validation error

KEY CONCEPTS - REGULARIZATION, BIAS AND VARIANCE

1. Learning curves may be useful, but often very noisy, messy and inconclusive

1. High Bias:

- Generally means the model is too simple
 - Try additional features
 - Try different features (higher degree polynomial)
 - Check your regularization parameter (is it too high?)

1. High Variance:

- Try a smaller set of features
 - Either less number of features or a lower degree polynomial
- Check your regularization parameter (is it too low?)