Lecture 6: Multiple Linear Regression, Polynomial Regression and Model Selection

CS109A Introduction to Data Science

Pavlos Protopapas and Kevin Rader

Announcements

Section: Friday 1:30-2:45pm : @ MD 123 (only this Friday)

A-section: Today: 5:00-6:30pm @60 Oxford str. Room 330

Mixer: Today 7:30pm @IACS lobby

Regrade requests:

HW1 grades are released. For regrade requests email the helpline with subject line **Regrade HW1: Grader=johnsmith** within **48 hours** of the grade release.

Lecture Outline

Multiple Linear Regression:

- Collinearity
- Hypothesis Testing
- Categorical Predictors
- Interaction Terms

Polynomial Regression

Generalized Polynomial Regression

Overfitting

Model Selection

- Exhaustive Selection
- Forward/Backward

AIC

Cross Validation

MLE

Multiple Linear Regression

Multiple Linear Regression

If you have to guess someone's height, would you rather be told

- Their weight, only
- Their weight and gender
- Their weight, gender, and income
- Their weight, gender, income, and favorite number

Of course, you'd always want as much data about a person as possible. Even though height and favorite number may not be strongly related, at worst you could just ignore the information on favorite number. We want our models to be able to take in lots of data as they make their predictions.

Response vs. Predictor Variables

X
predictors
features
covariates

Y
outcome
response variable
dependent variable

TV	radio	newspaper	sales
230.1	37.8	69.2	22.1
44.5	39.3	45.1	10.4
17.2	45.9	69.3	9.3
151.5	41.3	58.5	18.5
180.8	10.8	58.4	12.9

n observations

p predictors

Multilinear Models

In practice, it is unlikely that any response variable Y depends solely on one predictor x. Rather, we expect that is a function of multiple predictors $f(X_1, \ldots, X_J)$. Using the notation we introduced last lecture,

$$Y = y_1, \ldots, y_n, \quad X = X_1, \ldots, X_J \text{ and } X_j = x_{1j}, \ldots, x_{ij}, \ldots, x_{nj}$$

In this case, we can still assume a simple form for f -a multilinear form:

$$Y = f(X_1, \ldots, X_J) + \epsilon = \beta_0 + \beta_1 X_1 + \beta_2 X_2 + \ldots + \beta_J X_J + \epsilon$$

Hence, $\hat{f}$, has the form

$$\hat{Y} = \hat{f}(X_1, \ldots, X_J) + \epsilon = \hat{\beta}_0 + \hat{\beta}_1 X_1 + \hat{\beta}_2 X_2 + \ldots + \hat{\beta}_J X_J + \epsilon$$

Multiple Linear Regression

Again, to fit this model means to compute $\hat{\beta}_0, \dots, \hat{\beta}_J$ or to minimize a loss function; we will again choose the MSE as our loss function.

Given a set of observations,

$$\{(x_{1,1}, \dots, x_{1,J}, y_1), \dots (x_{n,1}, \dots, x_{n,J}, y_n)\},$$

the data and the model can be expressed in vector notation,

$$\mathbf{Y} = \begin{pmatrix} y_1 \\ \vdots \\ y_n \end{pmatrix} \qquad \mathbf{X} = \begin{pmatrix} 1 & x_{1,1} & \dots & x_{1,J} \\ 1 & x_{2,1} & \dots & x_{2,J} \\ \vdots & \vdots & \ddots & \vdots \\ 1 & x_{n,1} & \dots & x_{n,J} \end{pmatrix} \qquad \beta = \begin{pmatrix} \beta_0 \\ \beta_1 \\ \vdots \\ \beta_J \end{pmatrix}$$

Multiple Linear Regression

The model takes a simple algebraic form:

$$\mathbf{Y} = \mathbf{X}\beta + \epsilon$$

Thus, the MSE can be expressed in vector notation as

$$\text{MSE}(\beta) = \frac{1}{\text{n}}\|\boldsymbol{Y} - \boldsymbol{X}\boldsymbol{\beta}\|^2$$

Minimizing the MSE using vector calculus yields,

$$\widehat{\boldsymbol{\beta}} = \left(\mathbf{X}^\top\mathbf{X}\right)^{-1}\mathbf{X}^\top\mathbf{Y} = \underset{\boldsymbol{\beta}}{\operatorname{argmin}}\ \text{MSE}(\boldsymbol{\beta}).$$

Collinearity

Collinearity refers to the case in which two or more predictors are correlated (related).

We will re-visit collinearity in the next lectures, but for now we want to examine how does collinearity affects our confidence on the coefficients and consequently on the importance of those coefficients.

First let's look some examples:

Collinearity

Three individual models

TV

Coef.	Std.Err.	t	P>|t|	[0.025	0.975]
6.679	0.478	13.957	2.804e-31	5.735	7.622
0.048	0.0027	17.303	1.802e-41	0.042	0.053

RADIO

Coef.	Std.Err.	t	P>|t|	[0.025	0.975]
9.567	0.553	17.279	2.133e-41	8.475	10.659
0.195	0.020	9.429	1.134e-17	0.154	0.236

NEWS

Coef.	Std.Err.	t	P>|t|	[0.025	0.975]
11.55	0.576	20.036	1.628e-49	10.414	12.688
0.074	0.014	5.134	6.734e-07	0.0456	0.102

One model

	Coef.	Std.Err.	t	P>|t|	[0.025	0.975]
β_0	2.602	0.332	7.820	3.176e-13	1.945	3.258
β_{TV}	0.046	0.0015	29.887	6.314e-75	0.043	0.049
β_{RADIO}	0.175	0.0094	18.576	4.297e-45	0.156	0.194
β_{NEWS}	0.013	0.028	2.338	0.0203	0.008	0.035

Collinearity

Collinearity refers to the case in which two or more predictors are correlated (related).

We will re-visit collinearity in the next lectures, but for now we want to examine how does collinearity affects our confidence on the coefficients and consequently on the importance of those coefficients.

Assuming uncorrelated noise then we can show:

$$\mathrm{Cov}(\hat{\beta}) = \hat{\sigma}^2 (X^T X)^{-1}$$

Finding Significant Predictors: Hypothesis Testing

For checking the significance of linear regression coefficients:

1.we set up our hypotheses H_0:

$$H_0 : \beta_0 = \beta_1 = \ldots = \beta_J = 0 \qquad \textbf{(Null)}$$

$$H_1 : \beta_j \neq 0, \text{ for at least one } j \qquad \textbf{(Alternative)}$$

2. we choose the *F*-stat to evaluate the null hypothesis,

$$F = \frac{\text{explained variance}}{\text{unexplained variance}}$$

Finding Significant Predictors: Hypothesis Testing

3. we can compute the *F*-stat for linear regression models by

$$F = \frac{(TSS - RSS)/J}{RSS/(n - J - 1)}, \quad TSS = \sum_i (y_i - \bar{y})^2, \; RSS = \sum_i (y_i - \hat{y}_i)^2$$

4. If $F = 1$ we consider this evidence for H_0; if $F > 1$, we consider this evidence against H_0.

Qualitative Predictors

So far, we have assumed that all variables are quantitative. But in practice, often some predictors are **qualitative**.

Example: The Credit data set contains information about balance, age, cards, education, income, limit , and rating for a number of potential customers.

Income	Limit	Rating	Cards	Age	Education	Gender	Student	Married	Ethnicity	Balance
14.890	3606	283	2	34	11	Male	No	Yes	Caucasian	333
106.02	6645	483	3	82	15	Female	Yes	Yes	Asian	903
104.59	7075	514	4	71	11	Male	No	No	Asian	580
148.92	9504	681	3	36	11	Female	No	No	Asian	964
55.882	4897	357	2	68	16	Male	No	Yes	Caucasian	331

Qualitative Predictors

If the predictor takes only two values, then we create an **indicator** or **dummy variable** that takes on two possible numerical values.

For example for the gender, we create a new variable:

$$x_i = \begin{cases} 1 & \text{if } i\,\text{th person is female} \\ 0 & \text{if } i\,\text{th person is male} \end{cases}$$

We then use this variable as a predictor in the regression equation.

$$y_i = \beta_0 + \beta_1 x_i + \epsilon_i = \begin{cases} \beta_0 + \beta_1 + \epsilon_i & \text{if } i\,\text{th person is female} \\ \beta_0 + \epsilon_i & \text{if } i\,\text{th person is male} \end{cases}$$

Qualitative Predictors

Question: What is interpretation of β_0 and β_1?

- β_0 is the average credit card balance among males,
- $\beta_0 + \beta_1$ is the average credit card balance among females,
- and β_1 the average difference in credit card balance between females and males.

Exercise: Calculate β_0 and β_1 for the Credit data.

You should find $\beta_0 \sim \$509, \beta_1 \sim \$19$

More than two levels: One hot encoding

Often, the qualitative predictor takes more than two values (e.g. ethnicity in the credit data).

In this situation, a single dummy variable cannot represent all possible values.

We create additional dummy variable as:

$$x_{i,1} = \begin{cases} 1 & \text{if } i \text{th person is Asian} \\ 0 & \text{if } i \text{th person is not Asian} \end{cases}$$

$$x_{i,2} = \begin{cases} 1 & \text{if } i \text{th person is Caucasian} \\ 0 & \text{if } i \text{th person is not Caucasian} \end{cases}$$

More than two levels: One hot encoding

We then use these variables as predictors, the regression equation becomes:

$$y_i = \beta_0 + \beta_1 x_{i,1} + \beta_2 x_{i,2} + \epsilon_i = \begin{cases} \beta_0 + \beta_1 + \epsilon_i & \text{if } i\,\text{th person is Asian} \\ \beta_0 + \beta_2 + \epsilon_i & \text{if } i\,\text{th person is Caucasian} \\ \beta_0 + \epsilon_i & \text{if } i\,\text{th person is AfricanAmerican} \end{cases}$$

Question: What is the interpretation of $\beta_0, \beta_1, \beta_2$

Beyond linearity

In the Advertising data, we assumed that the effect on sales of increasing one advertising medium is independent of the amount spent on the other media.

If we assume linear model then the average effect on sales of a one-unit increase in TV is always β_1, regardless of the amount spent on radio.

Synergy effect or **interaction effect** states that when an increase on the radio budget affects the effectiveness of the TV spending on sales.

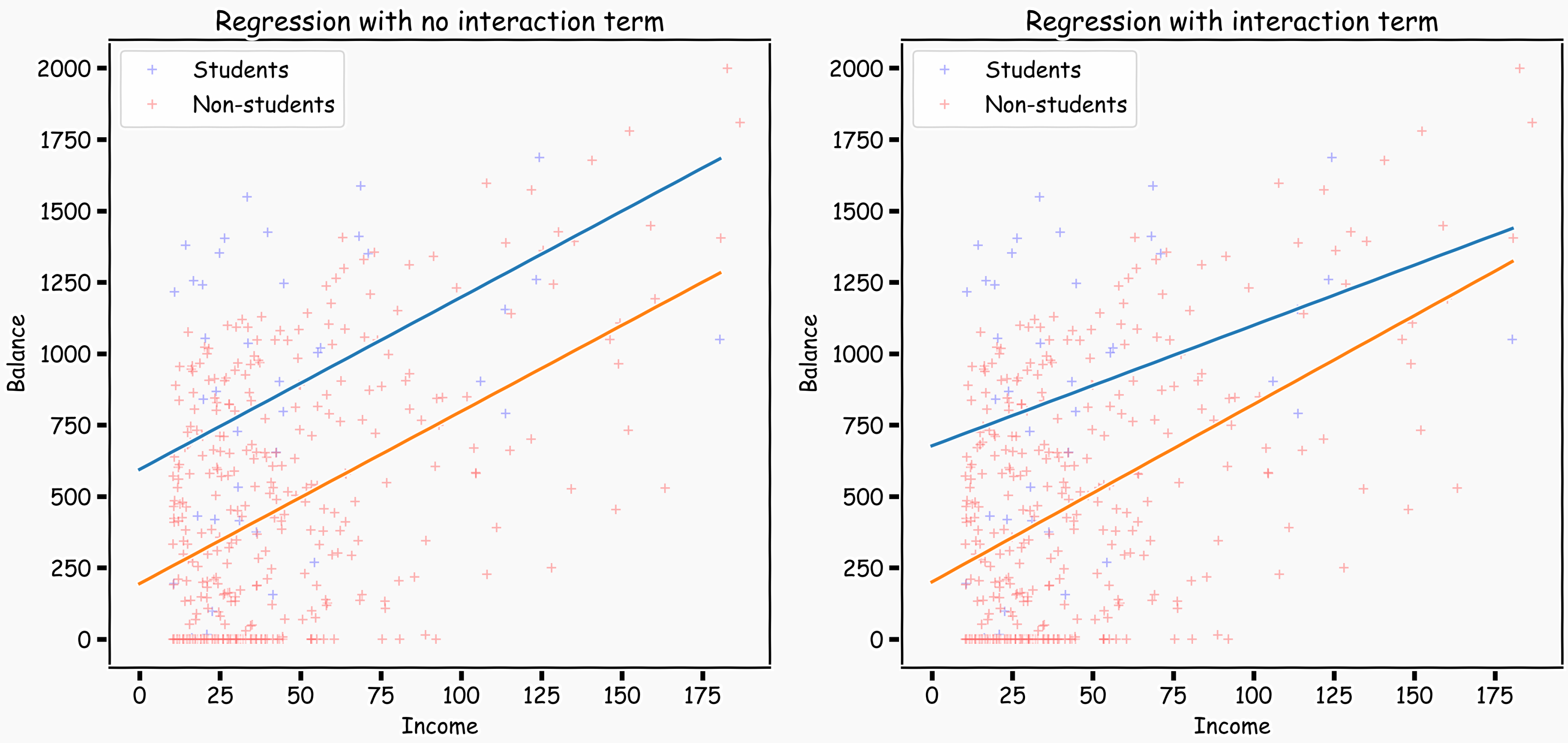
Beyond linearity

We change

$$Y = \beta_0 + \beta_1 X_1 + \beta_2 X_2 + \epsilon$$

To

$$Y = \beta_0 + \beta_1 X_1 + \beta_2 X_2 + \boxed{\beta_3 X_1 X_2} + \epsilon$$



Question: Explain the plots above?

Predictors predictors predictors

We have a lot predictors!

Is it a problem?

- Yes: Computational Cost
- Yes: Overfitting

Wait there is more …

Polynomial Regression

Polynomial Regression

The simplest non-linear model we can consider, for a response *Y* and a predictor *X*, is a polynomial model of degree *M*,

$$y = \beta_0 + \beta_1 x + \beta_2 x^2 + \ldots + \beta_M x^M + \epsilon.$$

Just as in the case of linear regression with cross terms, polynomial regression is a special case of linear regression - we treat each x^m as a separate predictor. Thus, we can write:

$$\mathbf{Y} = \begin{pmatrix} y_1 \\ \vdots \\ y_n \end{pmatrix}, \qquad \mathbf{X} = \begin{pmatrix} 1 & x_1^1 & \ldots & x_1^M \\ 1 & x_2^1 & \ldots & x_2^M \\ \vdots & \vdots & \ddots & \vdots \\ 1 & x_n & \ldots & x_n^M \end{pmatrix}, \qquad \boldsymbol{\beta} = \begin{pmatrix} \beta_0 \\ \beta_1 \\ \vdots \\ \beta_M \end{pmatrix}.$$

Polynomial Regression

Design Matrix

$$\mathbf{Y} = \begin{pmatrix} y_1 \\ \vdots \\ y_n \end{pmatrix}, \qquad \mathbf{X} = \begin{pmatrix} 1 & x_1^1 & \ldots & x_1^M \\ 1 & x_2^1 & \ldots & x_2^M \\ \vdots & \vdots & \ddots & \vdots \\ 1 & x_n & \ldots & x_n^M \end{pmatrix}, \qquad \boldsymbol{\beta} = \begin{pmatrix} \beta_0 \\ \beta_1 \\ \vdots \\ \beta_M \end{pmatrix}.$$

Again, minimizing the MSE using vector calculus yields,

$$\widehat{\boldsymbol{\beta}} = \underset{\boldsymbol{\beta}}{\operatorname{argmin}} \operatorname{MSE}(\boldsymbol{\beta}) = \left(\mathbf{X}^\top \mathbf{X}\right)^{-1} \mathbf{X}^\top \mathbf{Y}.$$

Generalized Polynomial Regression

We can generalize polynomial models:

1. consider polynomial models with multiple predictors $\{X_1, \ldots, X_j\}$:

$$
\begin{aligned}
y = & \beta_0 + \beta_1 x_1 + \ldots + \beta_M x_1^M \\
& + \beta_{M+1} x_2 + \ldots + \beta_{2M} x_2^M \\
& + \ldots \\
& + \beta_{M(J-1)+1} x_J + \ldots + \beta_{MJ} x_J^M
\end{aligned}
$$

2. consider polynomial models with multiple predictors $\{X_1, X_2\}$ and cross terms:

$$
\begin{aligned}
y = & \beta_0 + \beta_1 x_1 + \ldots + \beta_M x_1^M \\
& + \beta_{1+M} x_2 + \ldots + \beta_{2M} x_2^M \\
& + \beta_{1+2M} (x_1 x_2) + \ldots + \beta_{3M} (x_1 x_2)^M
\end{aligned}
$$

Generalized Polynomial Regression

In each case, we consider each term x_j^m, and each cross term $x_1 x_2$, as a unique predictor and apply linear regression:

$$\widehat{\boldsymbol{\beta}} = \underset{\boldsymbol{\beta}}{\operatorname{argmin}} \operatorname{MSE}(\boldsymbol{\beta}) = \left(\mathbf{X}^\top \mathbf{X}\right)^{-1} \mathbf{X}^\top \mathbf{Y}.$$

$$\operatorname{Cov}(\hat{\beta}) = \hat{\sigma}^2 (X^T X)^{-1}$$

Model Selection

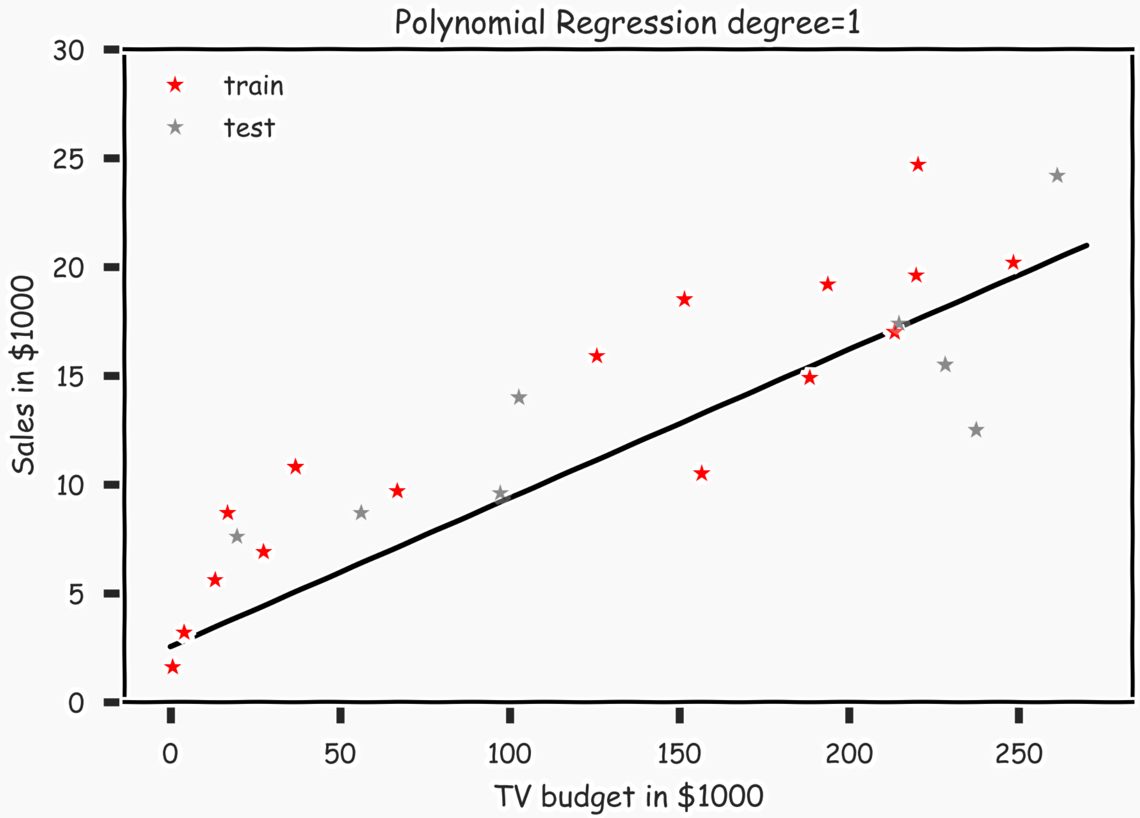
Model selection is the application of a principled method to determine the complexity of the model, e.g. choosing a subset of predictors, choosing the degree of the polynomial model etc.

A strong motivation for performing model selection is to avoid **overfitting**, which we can happen when:

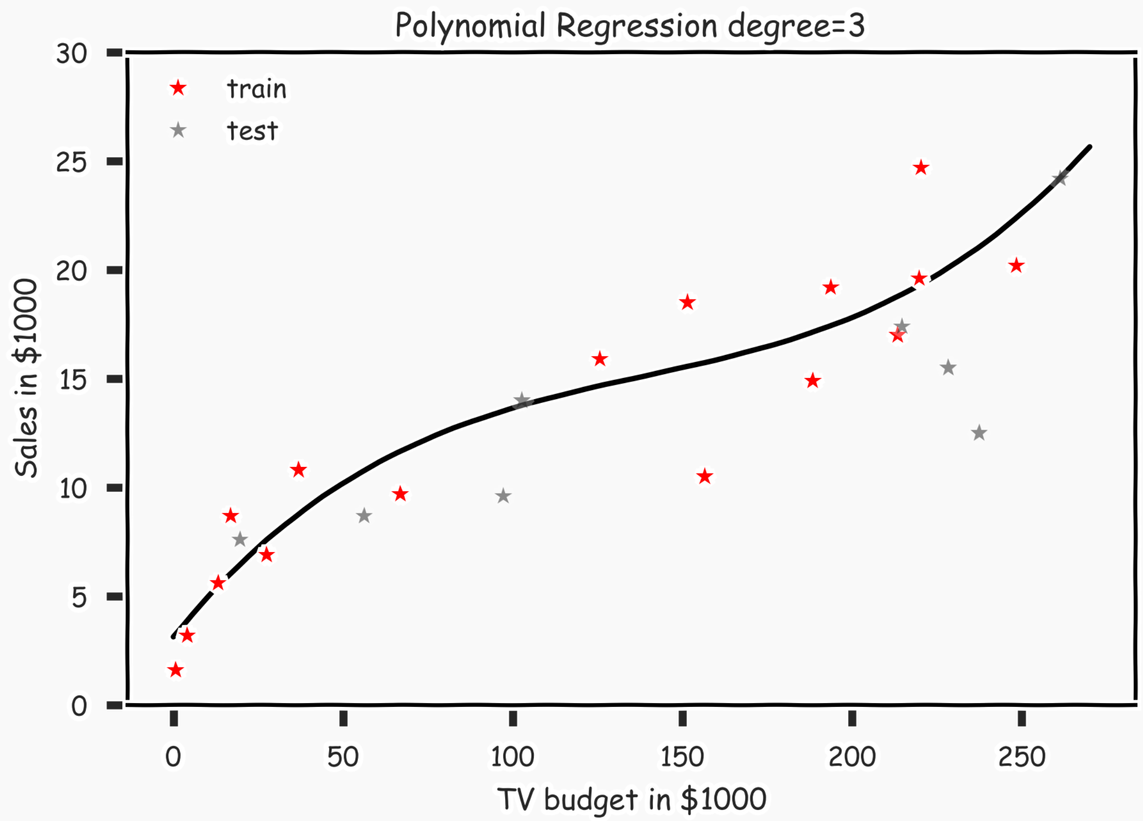
- there are too many predictors:
    - the feature space has high dimensionality
    - the polynomial degree is too high
    - too many cross terms are considered
- the coefficients values are too **extreme**

Overfitting

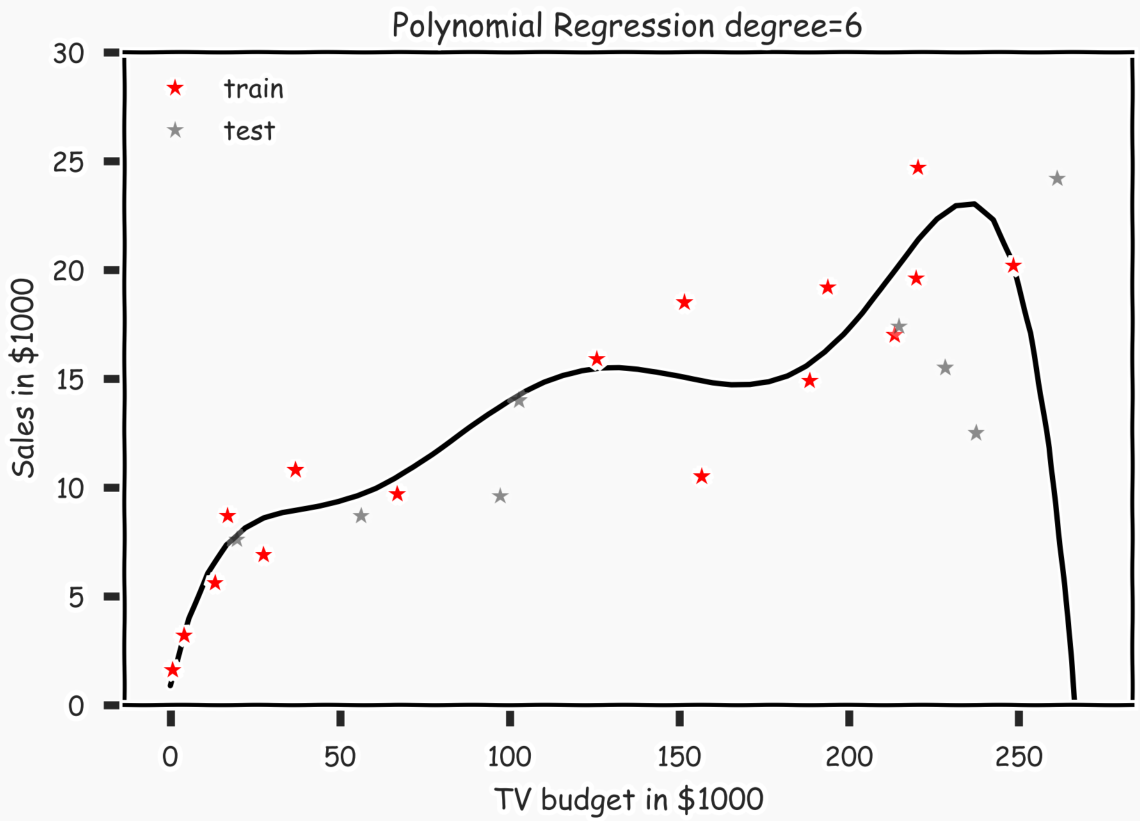
Overfitting



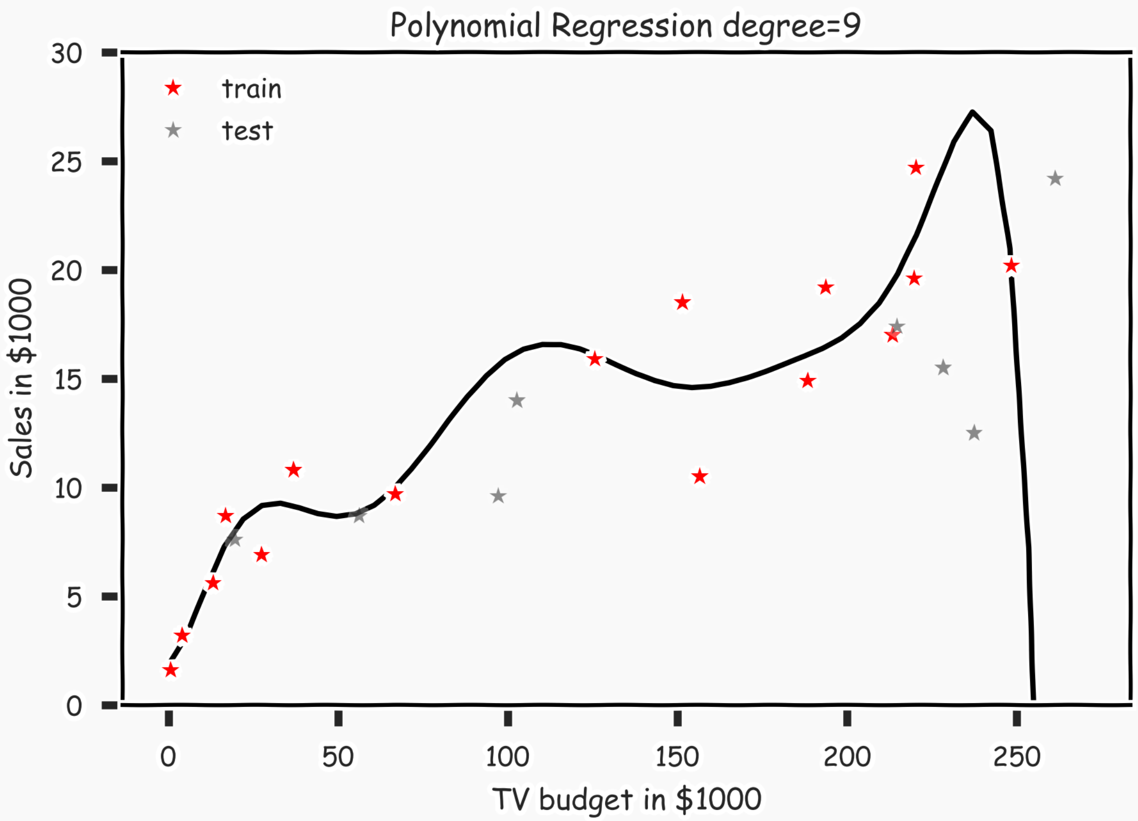
Overfitting



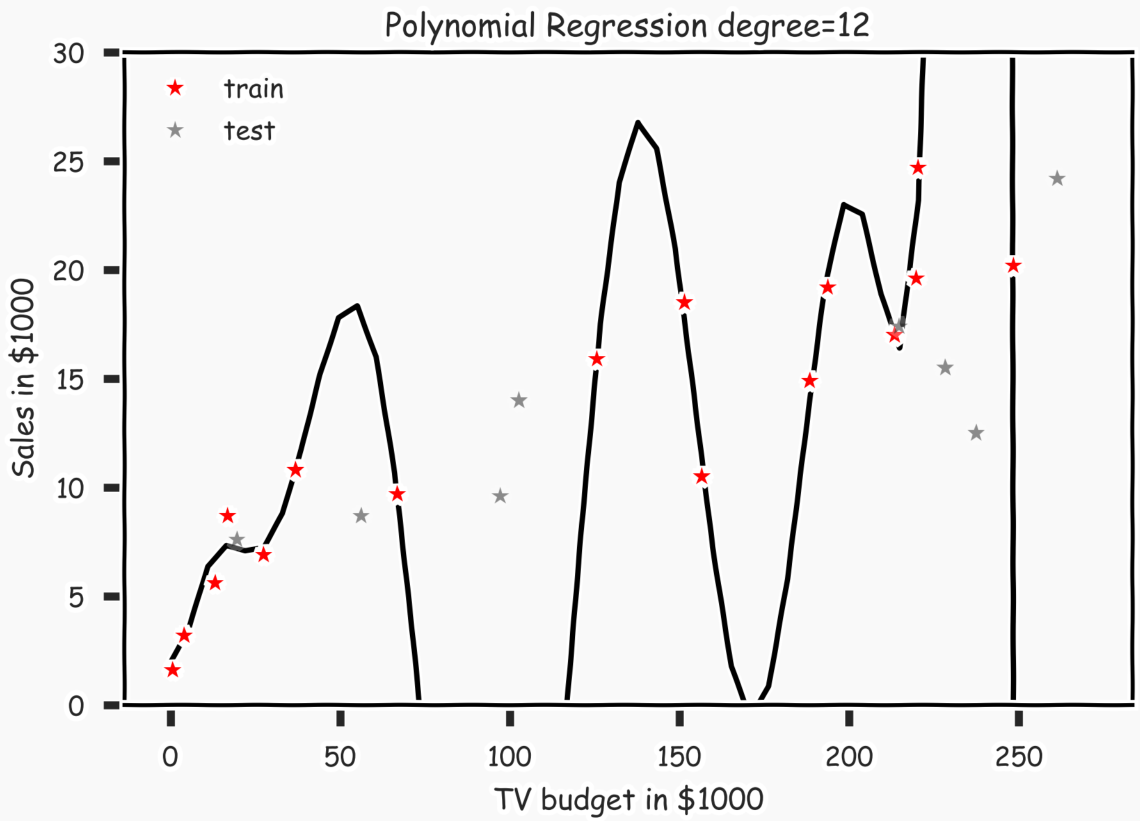
Overfitting



Overfitting



Overfitting

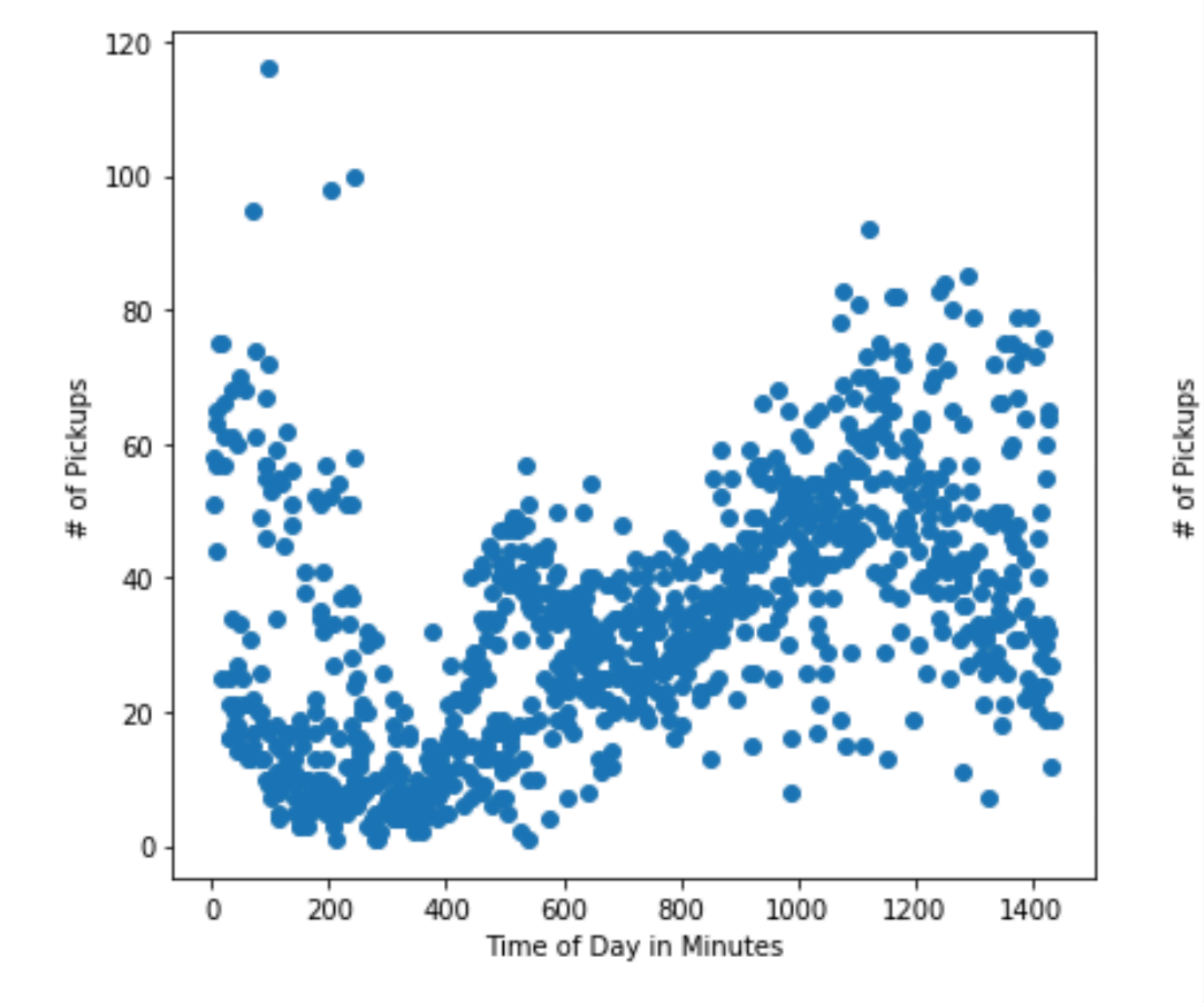


Overfitting

Definition

Overfitting is the phenomenon where the model is unnecessarily complex, in the sense that portions of the model captures the random noise in the observation, rather than the relationship between predictor(s) and response.

Overfitting causes the model to lose predictive power on new data.


120
100
80
60
40
20
0
# of Pickups
0
200
400
600
800
1000
1200
1400
Time of Day in Minutes

# of Pickups

Overfitting

As we saw, overfitting can happen when:

- there are too many predictors:
    - the feature space has high dimensionality
    - the polynomial degree is too high
    - too many cross terms are considered
- the coefficients values are too extreme

A sign of overfitting may be a high training R^2 or low MSE and unexpectedly poor testing performance.

Note: There is no 100% accurate test for overfitting and there is not a 100% effective way to prevent it. Rather, we may use multiple techniques in combination to prevent overfitting and various methods to detect it.

Model Selection

Exhaustive Selection

To find the optimal subset of predictors for modeling a response variable, we can:

- compute all possible subsets of $\{X_1, \ldots, X_J\}$
- evaluate all the models constructed from all the subsets of $\{X_1, \ldots, X_J\}$,
- find the model that optimizes some metric.

While straightforward, ***exhaustive selection*** is computationally infeasible, since $\{X_1, \ldots, X_J\}$ has 2^J number of possible subsets.

Instead, we will consider methods that iteratively build the optimal set of predictors.

Model selection

Model selection is the application of a principled method to determine the complexity of the model, e.g. choosing a subset of predictors, choosing the degree of the polynomial model etc.

Model selection typically consists of the following steps:

1. split the training set into two subsets: training and ***validation***
2. multiple models (e.g. polynomial models with different degrees) are fitted on the training set; each model is evaluated on the validation set
3. the model with the best validation performance is selected
4. the selected model is evaluated one last time on the testing set

Variable Selection: Forward

In ***forward selection***, we find an ‘optimal’ set of predictors by iterative building up our set.

1. Start with the empty set P_0, construct the null model M_0.
2. For $k = 1 \ldots J$:
   A. Let M_{k-1} be the model constructed from the best set of $k-1$ predictors, P_{k-1}.
   B. Select the predictor X_{n_k}, not in P_{k-1}, so that the model constructed from $P_k = X_{n_k} \cup P_{k-1}$ optimizes a fixed metric (this can be p-value, F-stat; validation MSE, R^2; or **AIC/BIC** on training set).
   C. Let M_k denote the model constructed from the optimal Pk.
3. Select the model M amongst $\{M_0, M_1, \ldots, M_J\}$ that optimizes a fixed metric (this can be validation MSE, R^2; or **AIC/BIC** on training set).
4. Evaluate the final model M on the testing set.

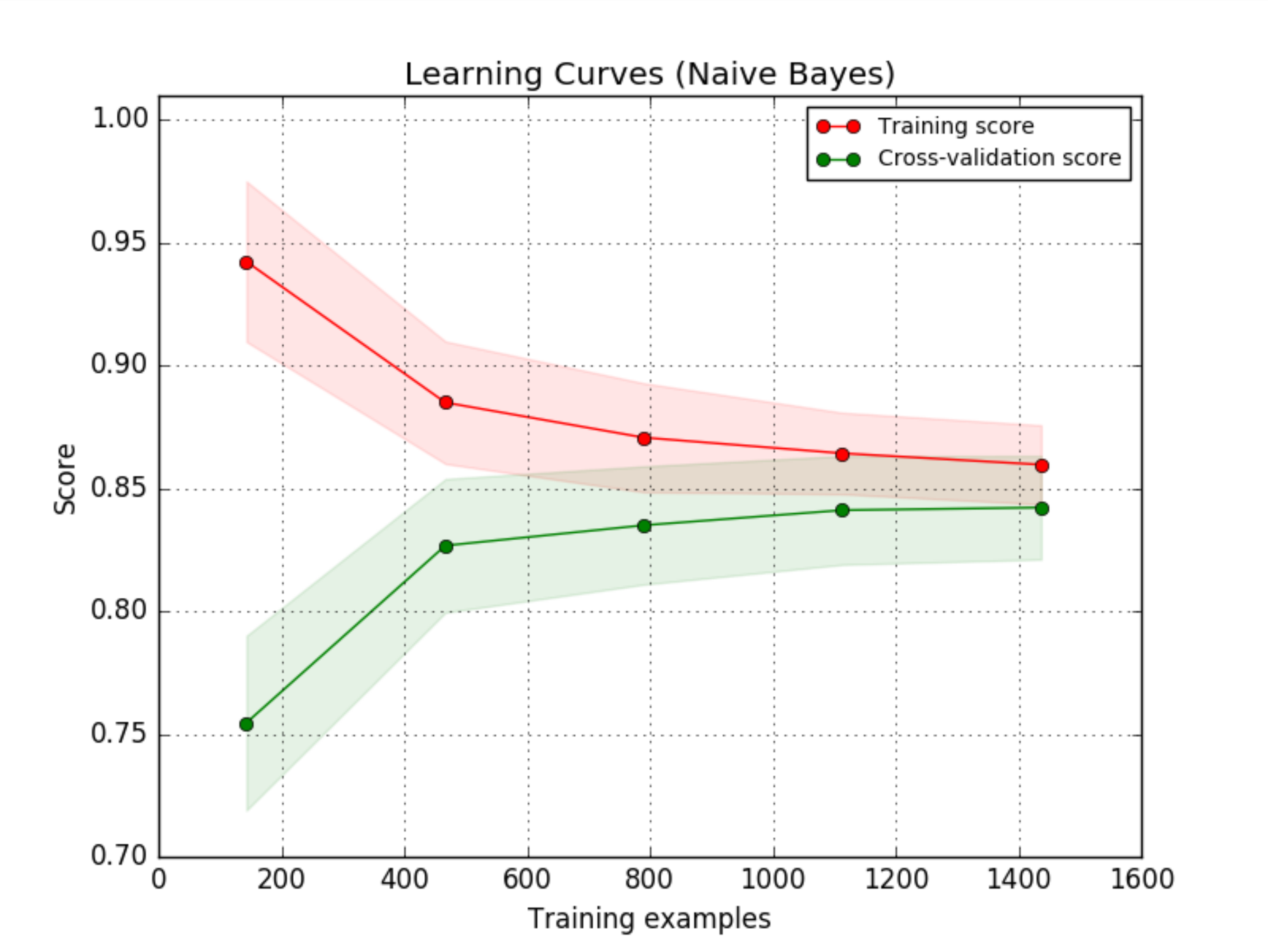
Stepwise Variable Selection Computational Complexity

How many models did we evaluate?

- 1st step, *J* **Models**
- 2nd step, *J-1* **Models** (add 1 predictor out of *J-1* possible)
- 3rd step, *J*-2 **Models** (add 1 predictor out of *J*-2 possible)
- …

$$O(J^2) \ll 2^J \text{ for large } J$$

AIC and BIC – value of training data



AIC and BIC

In the absence of training data (we may not want to use valuable data for validation)

We've mentioned using AIC/BIC to evaluate the explanatory powers of models. The following formulae can be used to calculate these criteria:

$$\mathrm{AIC} \approx 2n\ln(\mathrm{MSE}) + 2J$$

$$\mathrm{BIC} \approx 2n\ln(\mathrm{MSE}) + 2J\ln n$$

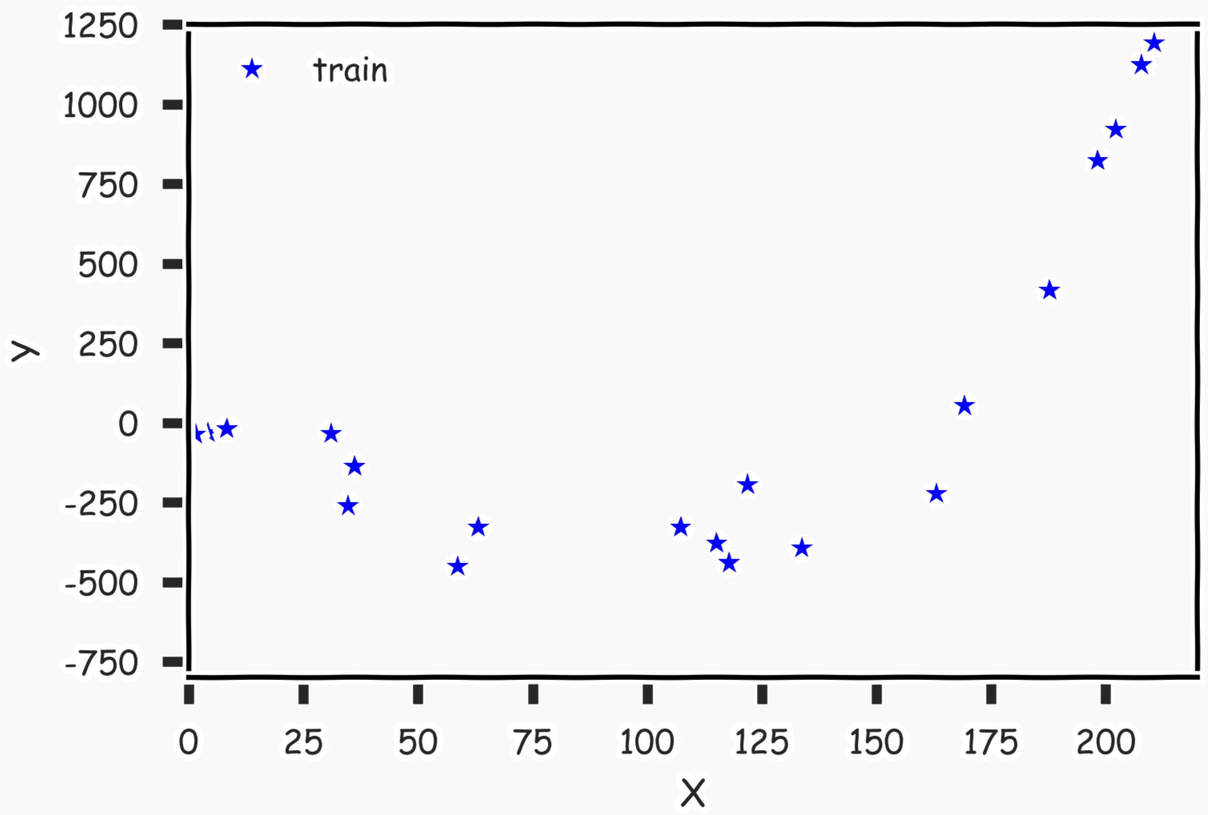
where *J* is the number of predictors in model.

Intuitively, AIC/BIC is a loss function that depends both on the predictive error, MSE, and the complexity of the model. We see that we prefer a model with few parameters and low MSE.

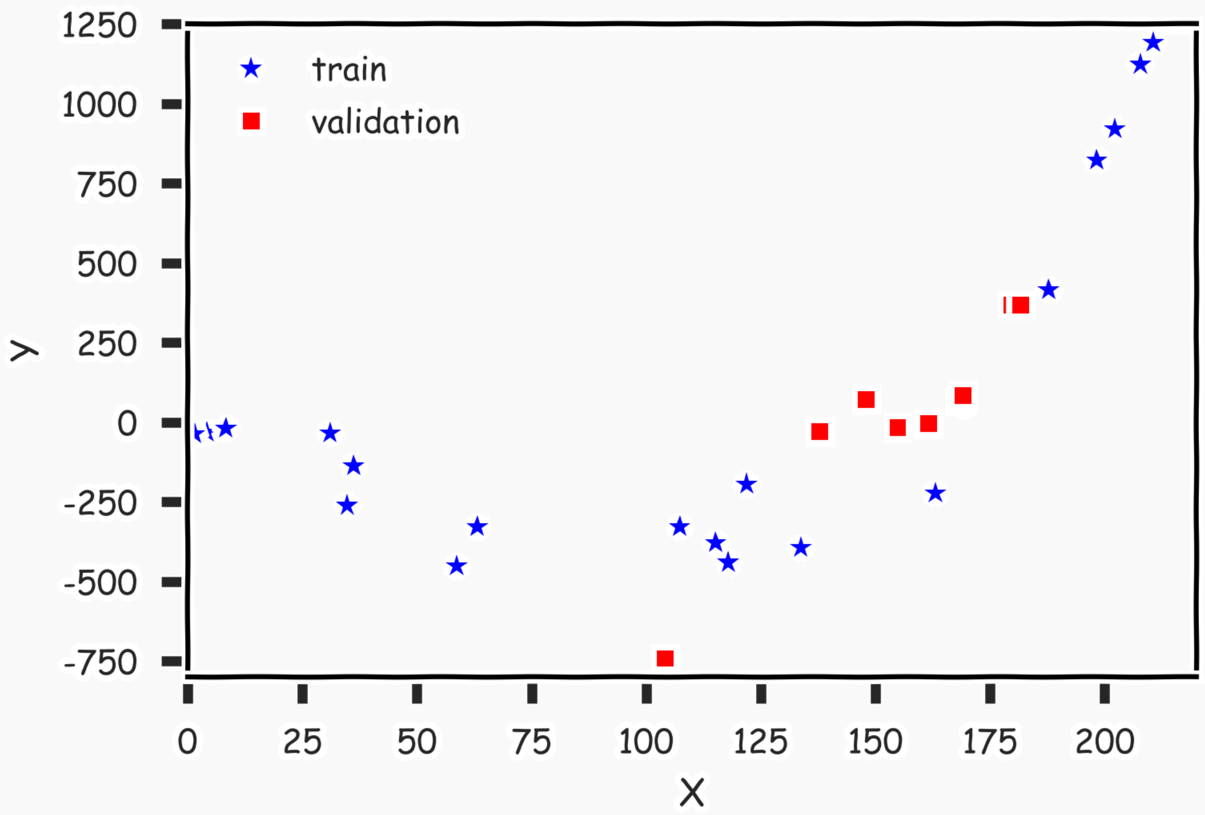
But why do the formulae look this way - what is the justification? We will cover all that in A-sec2 today

Cross Validation

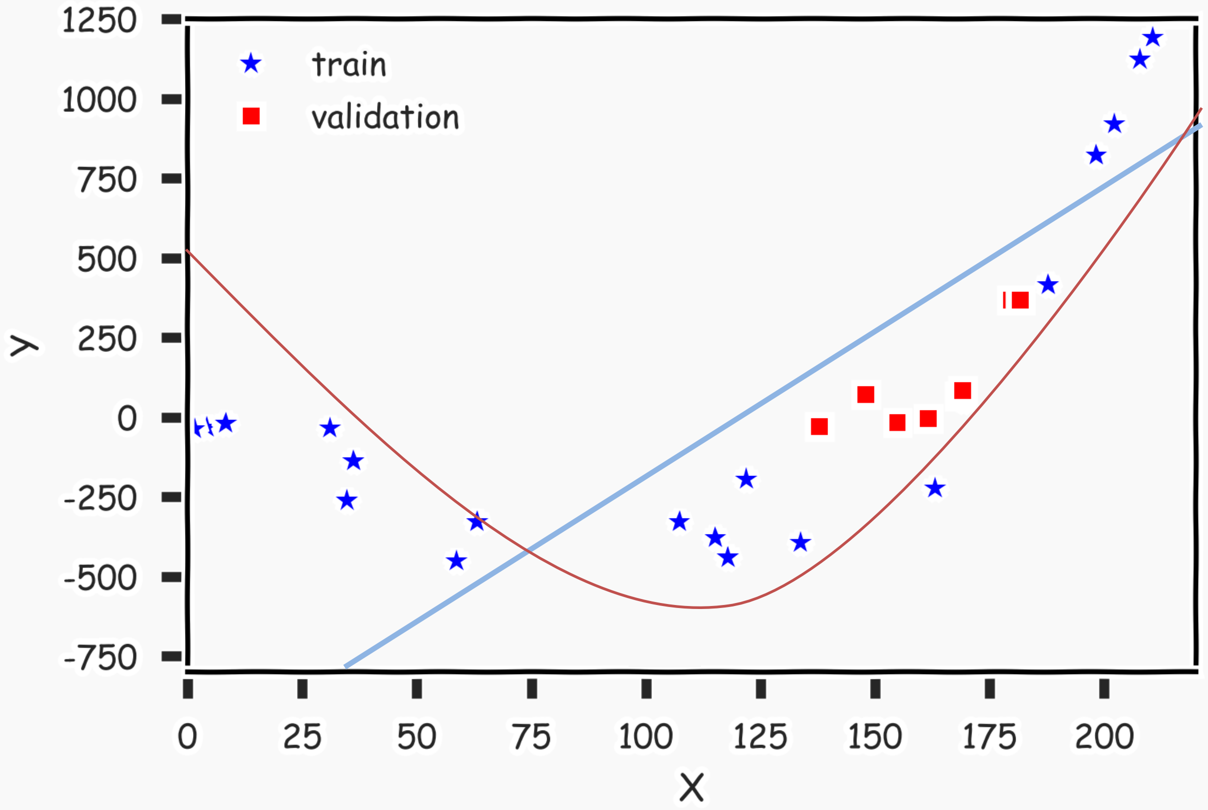
Cross Validation



Cross Validation



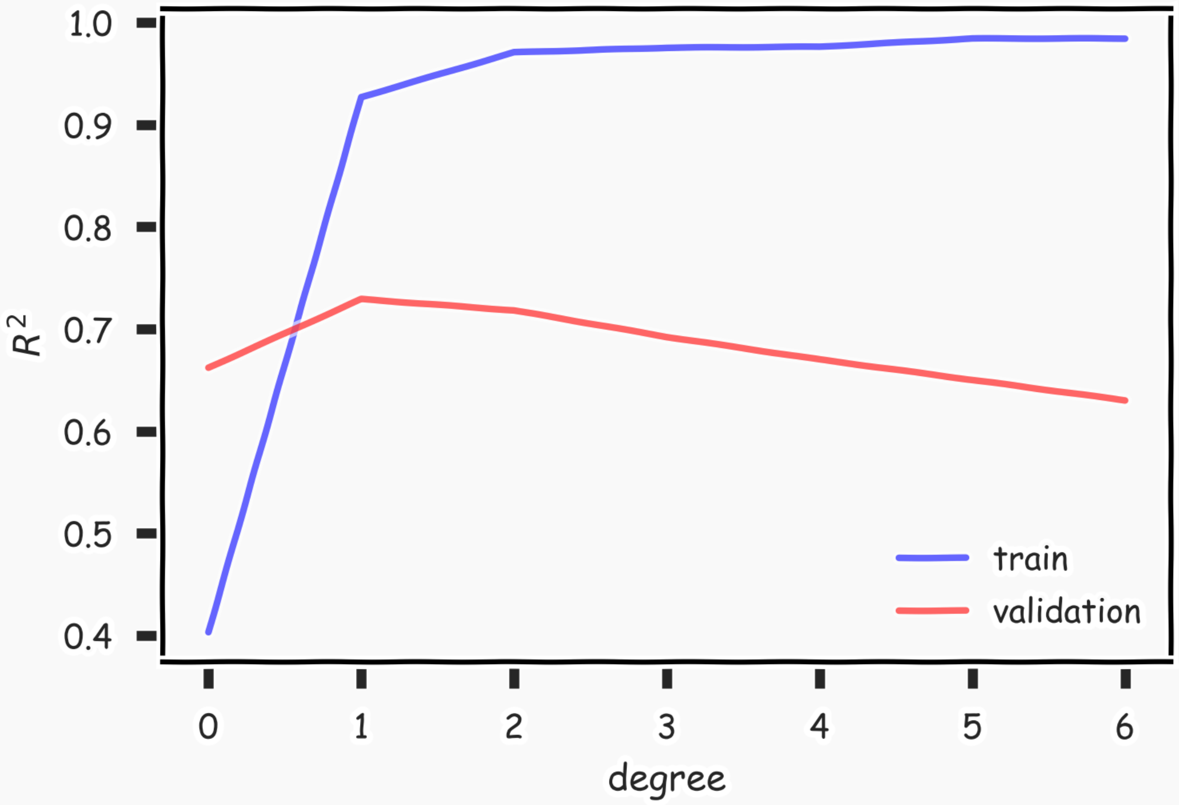
Cross Validation



Linear

Quadratic

Validation



Cross Validation: Motivation

Using a single validation set to select amongst multiple models can be problematic - **there is the possibility of overfitting to the validation set**.

One solution to the problems raised by using a single validation set is to evaluate each model on **multiple** validation sets and average the validation performance.

One can randomly split the training set into training and validation multiple times **but** randomly creating these sets can create the scenario where important features of the data never appear in our random draws.

Leave-One-Out

Given a data set $\{X_1, \ldots, X_n\}$, where each $\{X_1, \ldots, X_n\}$ contains J features.

To ensure that every observation in the dataset is included in at least one training set and at least one validation set, we create training/validation splits using the ***leave one out*** method:

- validation set: $\{X_i\}$
- training set: $X_{-1} = \{X_1, \ldots, X_{i-1}, X_{i+1}, \ldots, X_n\}$

for $i = 1, \ldots, n$:

We fit the model on each training set, denoted $\hat{f}_{X_{-1}}$, and evaluate it on the corresponding validation set, $\hat{f}_{X_{-1}}(X_i)$.

The ***cross validation score*** is the performance of the model averaged across all validation sets:

$$CV(\text{Model}) = \frac{1}{n}\sum_{i=1}^{n} L(\hat{f}_{X_{-1}}(X_i))$$

where L is a loss function.

K-Fold Cross Validation

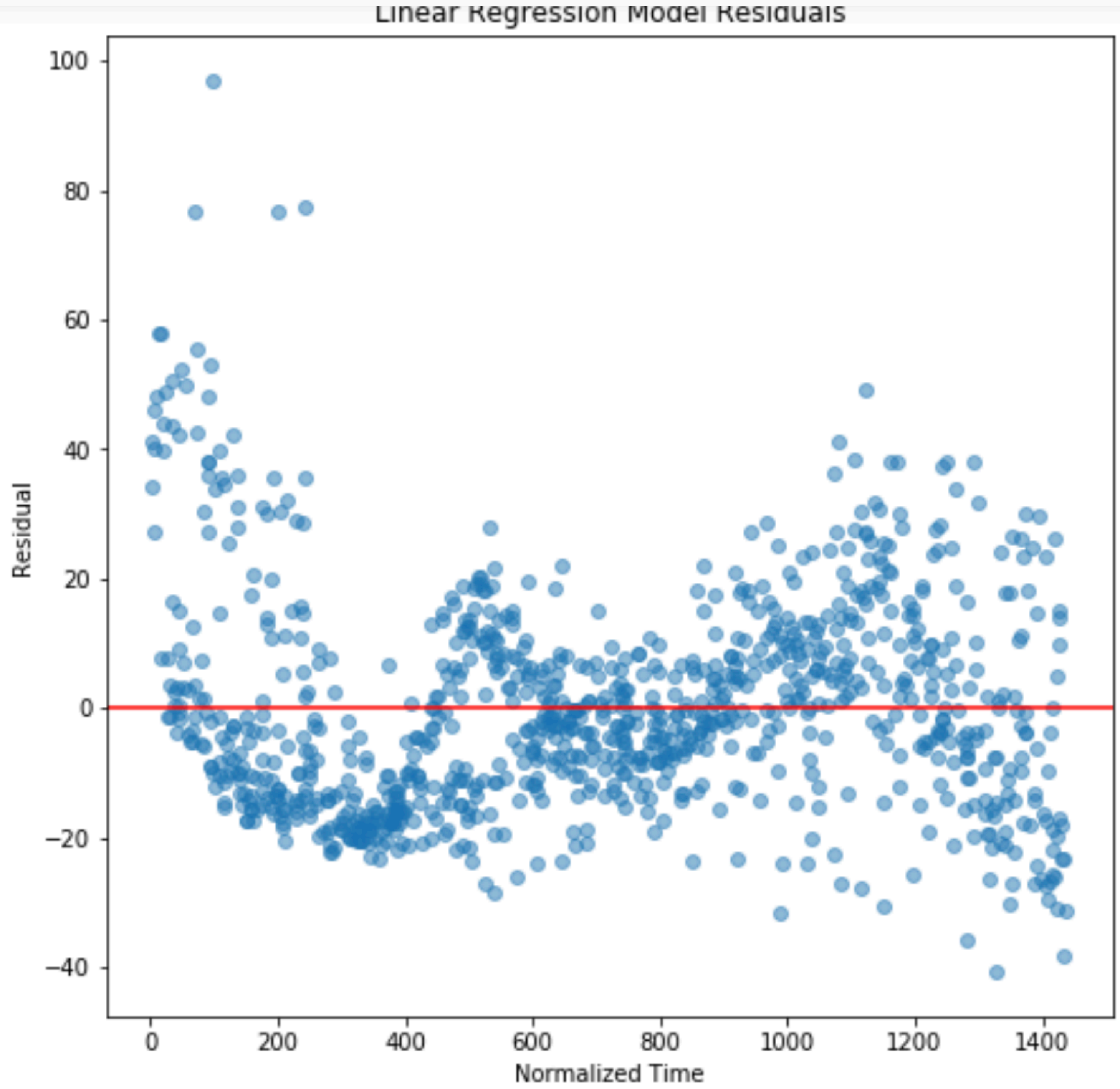
Rather than creating *n* number of training/validation splits, each time leaving one data point for the validation set, we can include more data in the validation set using **K-fold validation**:

- split the data into *K* uniformly sized chunks, $\{C_1, \dots, C_K\}$
- we create *K* number of training/validation splits, using one of the *K* chunks for validation and the rest for training.

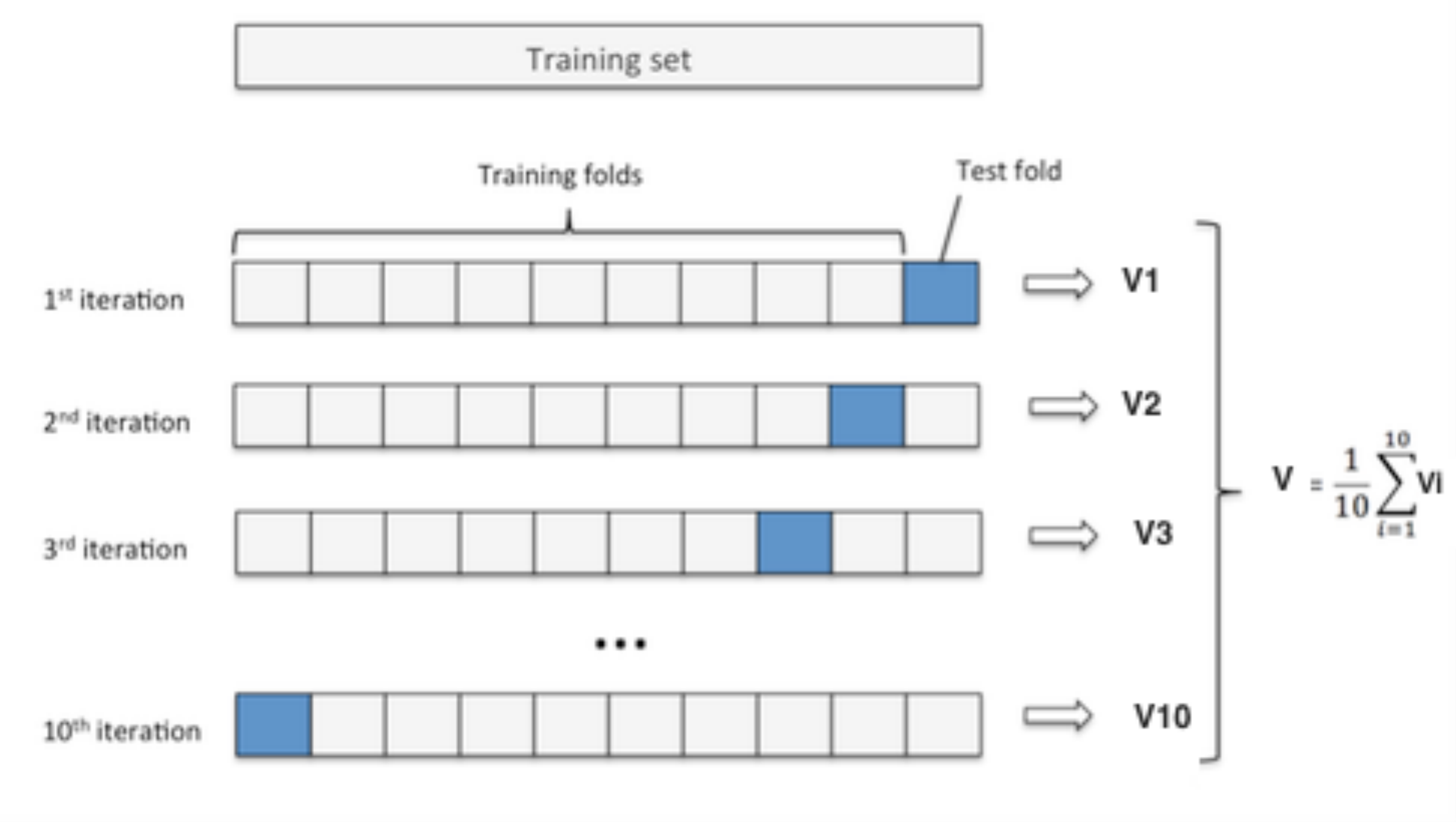
We fit the model on each training set, denoted $\hat{f}_{C_{-1}}$, and evaluate it on the corresponding validation set, $\hat{f}_{C_{-1}}(C_i)$. The ***cross validation is the performance*** of the model averaged across all validation sets:

$$CV(\text{Model}) = \frac{1}{K}\sum_{i=1}^{K} L(\hat{f}_{C_{-1}}(C_i))$$

where *L* is a loss function.


Linear Regression Model Residuals
Residual
100
80
60
40
20
0
−20
−40
0
200
400
600
800
1000
1200
1400
Normalized Time

Cross Validation



Predictor Selection: Cross Validation

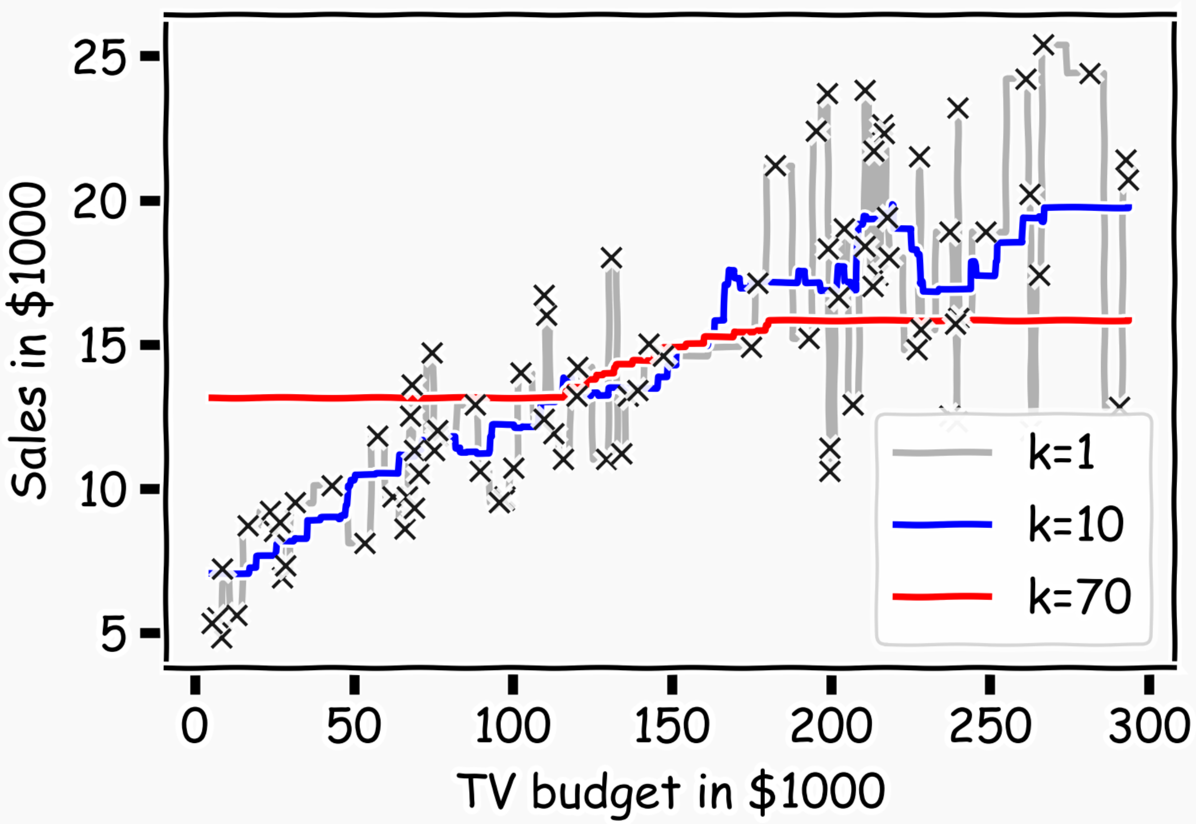
Question: What is the right ratio of train/validate/test, how do I choose K?

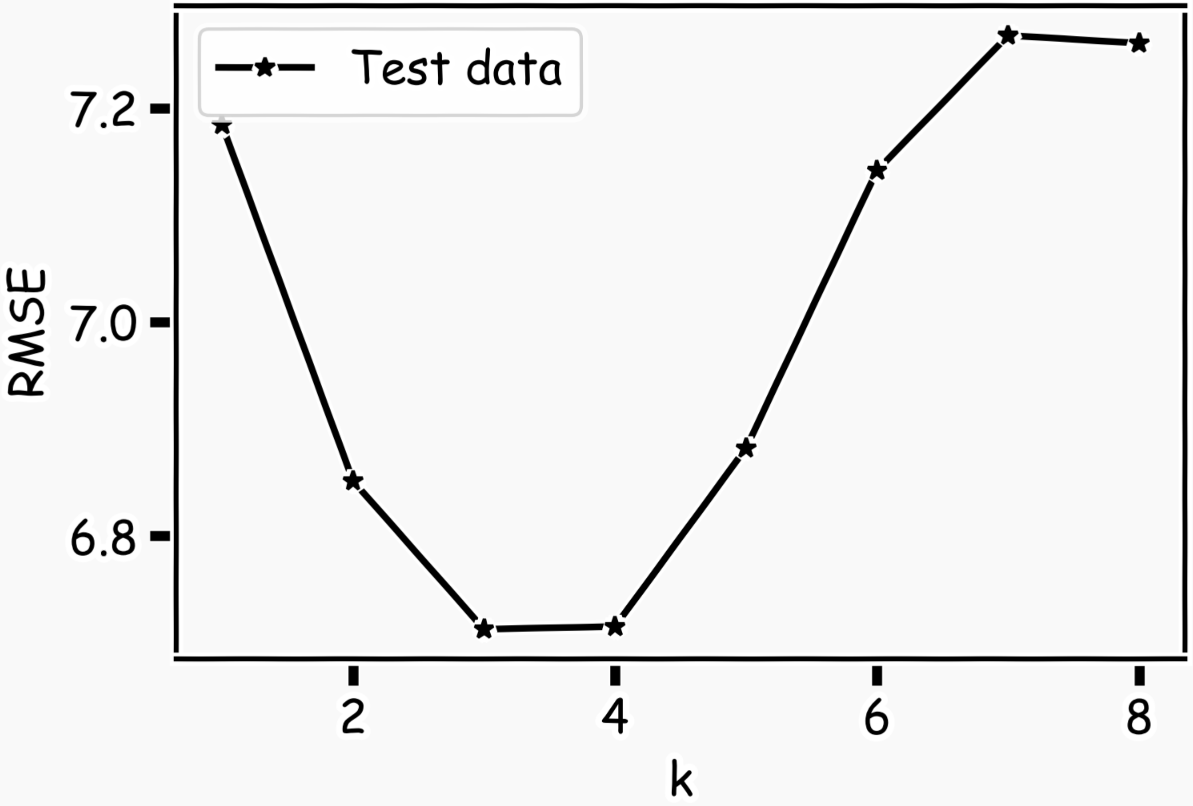
Question: What is the difference in multiple predictors and polynomial regression in model selection?

We can frame the problem of degree selection for polynomial models as a predictor selection problem:

which of the predictors $\{x, x^2, \dots, x^m\}$, should we select for modeling?

kNN Revisited





kNN Revisited

Recall our first simple, intuitive, non-parametric model for regression - the kNN model. We saw that it is vitally important to select an appropriate *k* for the data.

If the *k* is too small then the model is very sensitive to noise (since a new prediction is based on very few observed neighbors), and if the *k* is too large, the model tends towards making constant predictions.

A principled way to choose *k* is through K-fold cross validation.

Behind Ordinary Lease Squares, AIC, BIC

Likelihood Functions

Recall that our statistical model for linear regression in matrix notation is:

$$Y = X\beta + \epsilon$$

It is standard to suppose that $\epsilon \sim N(0, \sigma^2)$. In fact, in many analyses we have been making this assumption. Then,

$$y|\beta, x, \epsilon \sim \mathcal{N}(x\beta, \sigma^2)$$

Question: Can you see why?

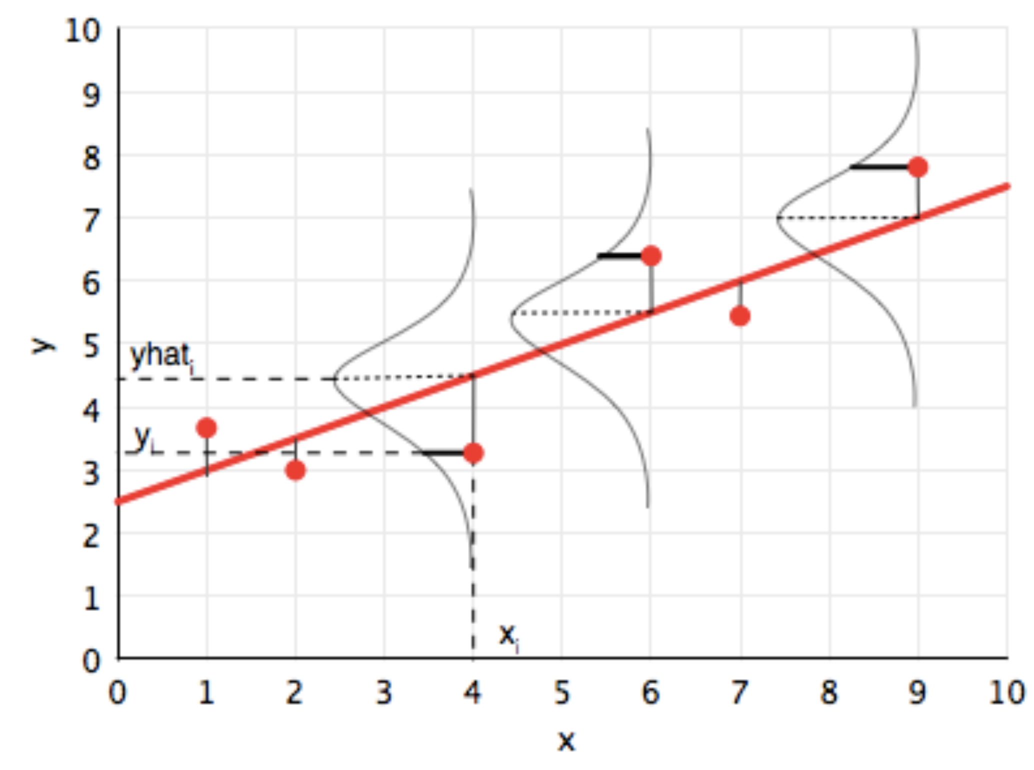
Note that $N(x\beta, \sigma^2)$ is naturally a function of the model parameters β, since the data is fixed.

Likelihood Functions

We call:

$$\mathcal{L}(\beta) = \mathcal{N}(x\beta, \sigma^2)$$

the **likelihood function**, as it gives the likelihood of the observed data for a chosen model $\boldsymbol{\beta}$.



Maximum Likelihood Estimators

Once we have a likelihood function, $\mathcal{L}(\boldsymbol{\beta})$, we have strong incentive to seek values of to maximize $\mathcal{L}$.

Can you see why?

The model parameters that maximizes $\mathcal{L}$ are called **maximum likelihood estimators (MLE)** and are denoted:

$$\boldsymbol{\beta}_{\text{MLE}} = \underset{\boldsymbol{\beta}}{\operatorname{argmax}}\, \mathcal{L}(\boldsymbol{\beta})$$

The model constructed with MLE parameters assigns the highest likelihood to the observed data.

Maximum Likelihood Estimators

But how does one maximize a likelihood function?

Fix a set of *n* observations of *J* predictors, **X**, and a set of corresponding response values, **Y**; consider a linear model $\boldsymbol{Y} = \boldsymbol{X\beta} + \epsilon$.

If we assume that $\epsilon \sim N(0, \sigma^2)$ then the likelihood for each observation is

$$\mathcal{L}_i(\boldsymbol{\beta}) = \mathcal{N}(y_i; \boldsymbol{\beta}^\top \boldsymbol{x}_i, \sigma^2)$$

and the likelihood for the entire set of data is

$$\mathcal{L}(\boldsymbol{\beta}) = \prod_{i=1}^{n} \mathcal{N}(y_i; \boldsymbol{\beta}^\top \boldsymbol{x}_i, \sigma^2)$$

Maximum Likelihood Estimators

Through some algebra, we can show that maximizing $\mathcal{L}(\boldsymbol{\beta})$, is equivalent to minimizing MSE:

$$\boldsymbol{\beta}_{MLE} = \underset{\boldsymbol{\beta}}{\operatorname{argmax}}\, \mathcal{L}(\boldsymbol{\beta}) = \underset{\boldsymbol{\beta}}{\operatorname{argmin}} \frac{1}{n}\sum_{i=1}^{n} |y_i - \boldsymbol{\beta}^\top \boldsymbol{x}_i|^2 = \underset{\boldsymbol{\beta}}{\operatorname{argmin}}\, MSE$$

Minimizing MSE or RSS is called **ordinary least squares**.

Information Criteria Revisited

Using the likelihood function, we can reformulate the information criteria metrics for model fitness in very intuitive terms.

For both AIC and BIC, we consider the likelihood of the data under the MLE model against the number of explanatory variables used in the model:

$$g(J) - \mathcal{L}(\boldsymbol{\beta}_{MLE})$$

where g is a function of the number of predictors *J*. Individually,

In the formulae we'd been using for AIC/BIC, we approximate $\mathcal{L}(\boldsymbol{\beta})$, using the MSE.