MACHINE LEARNING - I

UNIT # 3

1

TODAY'S AGENDA

- Recap of the previous lecture
- Subset Selection (Cont'd)
- Handling Categorical Variables (dummy variable vs one-hot encoding)
- Hold out and Cross Validation Methods
- AIC, BIC and Adjusted R2
- K Nearest Neighbor Regression

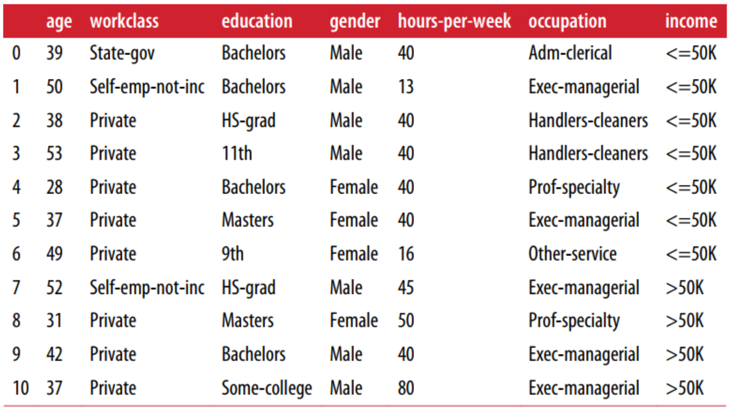
2

RECAP

- Assessing model accuracy (RSE and R2)
- Using p-values to identify not so relevant attributes
- Feature selection using Forward and Backward selection methods
- Interaction and Polynomial Models

HANDLING CATEGORICAL FEATURES

- A particularly common type of feature is the categorical features (or discrete features) that are not numeric.
- Consider the dataset of adult incomes in the United States, derived from the 1994 census database. The task of the adult dataset is to predict whether a worker has an income of over $50,000 or under $50,000.



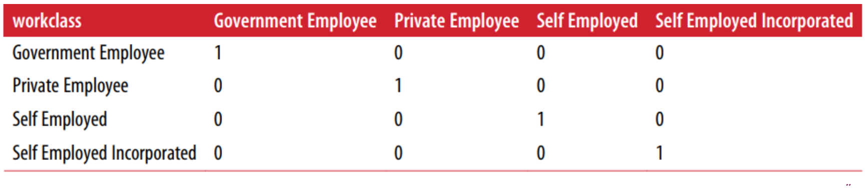
	age	workclass	education	gender	hours-per-week	occupation	income
0	39	State-gov	Bachelors	Male	40	Adm-clerical	<=50K
1	50	Self-emp-not-inc	Bachelors	Male	13	Exec-managerial	<=50K
2	38	Private	HS-grad	Male	40	Handlers-cleaners	<=50K
3	53	Private	11th	Male	40	Handlers-cleaners	<=50K
4	28	Private	Bachelors	Female	40	Prof-specialty	<=50K
5	37	Private	Masters	Female	40	Exec-managerial	<=50K
6	49	Private	9th	Female	16	Other-service	<=50K
7	52	Self-emp-not-inc	HS-grad	Male	45	Exec-managerial	>50K
8	31	Private	Masters	Female	50	Prof-specialty	>50K
9	42	Private	Bachelors	Male	40	Exec-managerial	>50K
10	37	Private	Some-college	Male	80	Exec-managerial	>50K

CATEGORICAL FEATURES (CONT'D)

- The features in this dataset include the workers' ages, how they are employed (self employed, private industry employee, government employee, etc.), their education, their gender, their working hours per week, occupation, and more.
- It would also be possible to predict the exact income, and make this a regression task.
- Age and hours-per-week are continuous features
- The workclass, education, sex, and occupation features are categorical

ONE-HOT ENCODING

- The most common way to represent categorical variables is using the one-hot encoding, also known as dummy variables. The approach replaces a categorical variable with one or more new features that can have the values 0 and 1.
- For the workclass feature we have 4 possible values: "Government Employee", "Private Employee", "Self Employed", and "Self Employed Incorporated".
- A feature is 1 if workclass for this person has the corresponding value and 0 otherwise.

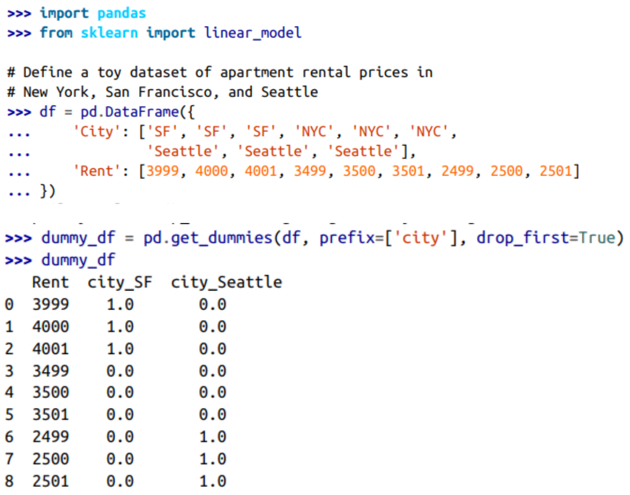


workclass	Government Employee	Private Employee	Self Employed	Self Employed Incorporated
Government Employee	1	0	0	0
Private Employee	0	1	0	0
Self Employed	0	0	1	0
Self Employed Incorporated	0	0	0	1

ONE HOT ENCODING VS DUMMY VARIABLES (IN STATISTICS)

- The one-hot encoding is similar, but not identical, to the dummy encoding used in statistics.
- Here, we encode each category with a different binary feature. In statistics, it is common to encode a categorical feature with k different possible values into k–1 features.
- This is done to avoid making the data matrix rank-deficient. Specifically, in the case of a linear regression model, a one hot encoding will cause the matrix of input data to become singular, meaning it cannot be inverted and the linear regression coefficients cannot be calculated using linear algebra. For these types of models a dummy variable encoding must be used instead.

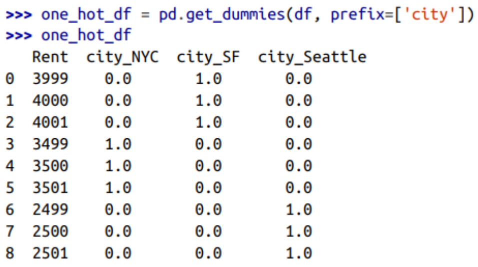
ONE HOT ENCODING VS DUMMY VARIABLES (CONT'D)



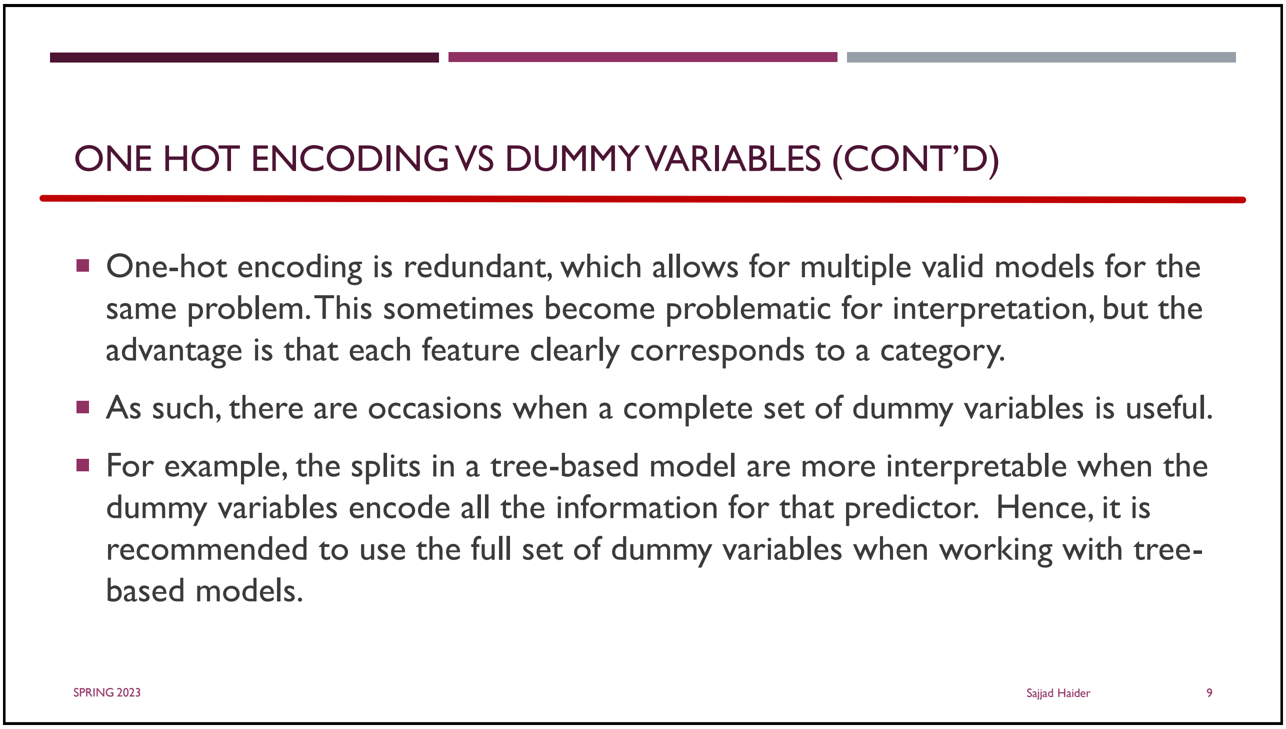
```
>>> import pandas
>>> from sklearn import linear_model

# Define a toy dataset of apartment rental prices in
# New York, San Francisco, and Seattle
>>> df = pd.DataFrame({
...     'City': ['SF', 'SF', 'SF', 'NYC', 'NYC', 'NYC',
...              'Seattle', 'Seattle', 'Seattle'],
...     'Rent': [3999, 4000, 4001, 3499, 3500, 3501, 2499, 2500, 2501]
... })

>>> dummy_df = pd.get_dummies(df, prefix=['city'], drop_first=True)
>>> dummy_df
   Rent  city_SF  city_Seattle
0  3999     1.0           0.0
1  4000     1.0           0.0
2  4001     1.0           0.0
3  3499     0.0           0.0
4  3500     0.0           0.0
5  3501     0.0           0.0
6  2499     0.0           1.0
7  2500     0.0           1.0
8  2501     0.0           1.0
```

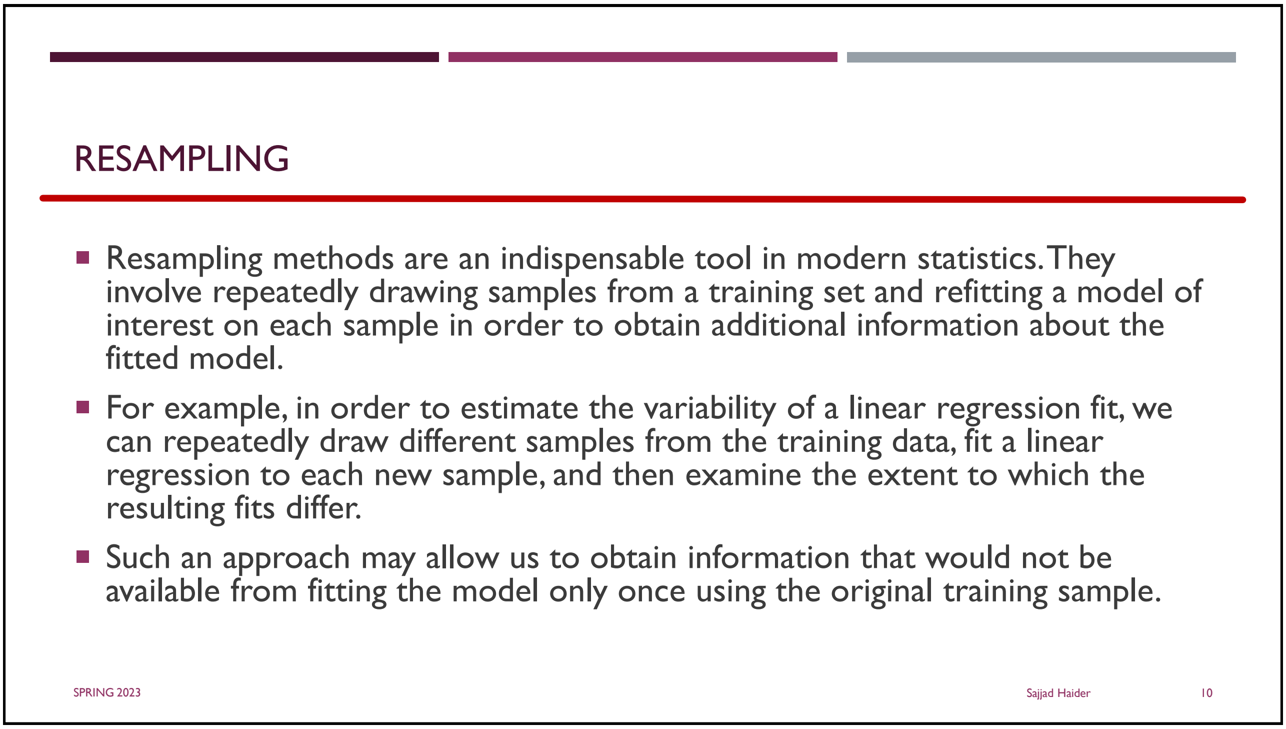


```
>>> one_hot_df = pd.get_dummies(df, prefix=['city'])
>>> one_hot_df
   Rent  city_NYC  city_SF  city_Seattle
0  3999       0.0      1.0           0.0
1  4000       0.0      1.0           0.0
2  4001       0.0      1.0           0.0
3  3499       1.0      0.0           0.0
4  3500       1.0      0.0           0.0
5  3501       1.0      0.0           0.0
6  2499       0.0      0.0           1.0
7  2500       0.0      0.0           1.0
8  2501       0.0      0.0           1.0
```



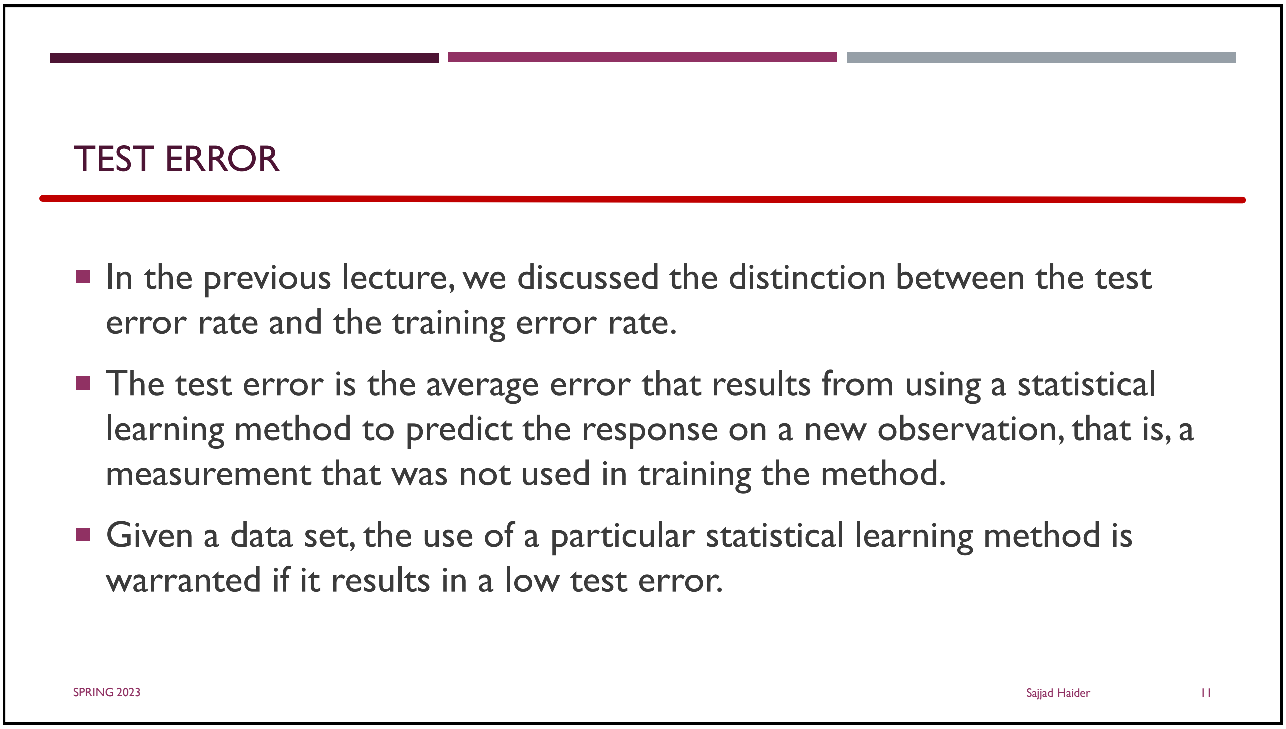
ONE HOT ENCODING VS DUMMY VARIABLES (CONT'D)

- One-hot encoding is redundant, which allows for multiple valid models for the same problem. This sometimes become problematic for interpretation, but the advantage is that each feature clearly corresponds to a category.
- As such, there are occasions when a complete set of dummy variables is useful.
- For example, the splits in a tree-based model are more interpretable when the dummy variables encode all the information for that predictor. Hence, it is recommended to use the full set of dummy variables when working with tree-based models.



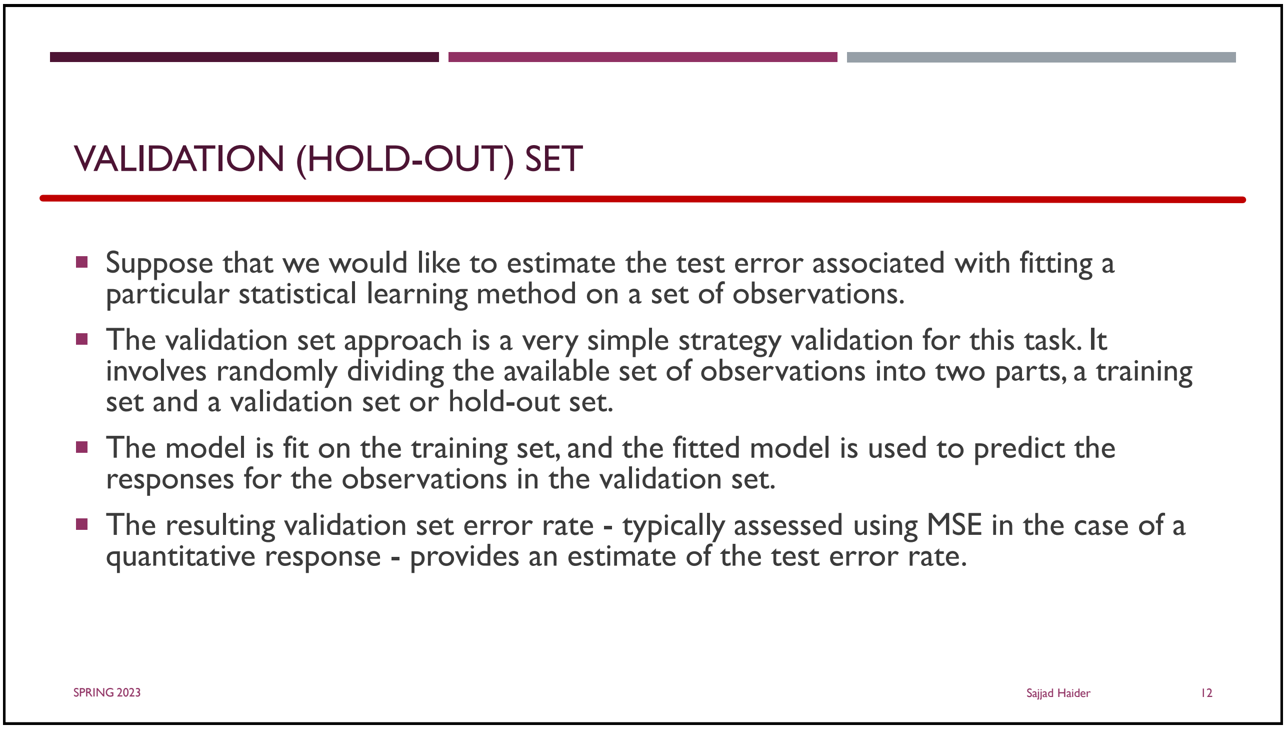
RESAMPLING

- Resampling methods are an indispensable tool in modern statistics. They involve repeatedly drawing samples from a training set and refitting a model of interest on each sample in order to obtain additional information about the fitted model.
- For example, in order to estimate the variability of a linear regression fit, we can repeatedly draw different samples from the training data, fit a linear regression to each new sample, and then examine the extent to which the resulting fits differ.
- Such an approach may allow us to obtain information that would not be available from fitting the model only once using the original training sample.



TEST ERROR

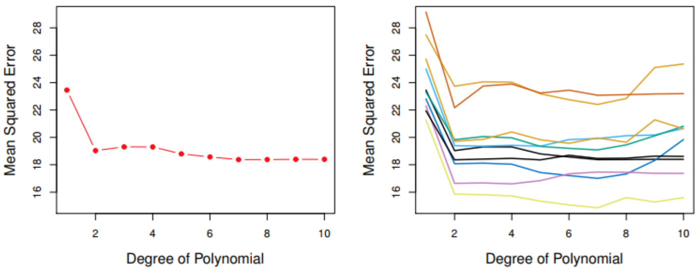
- In the previous lecture, we discussed the distinction between the test error rate and the training error rate.
- The test error is the average error that results from using a statistical learning method to predict the response on a new observation, that is, a measurement that was not used in training the method.
- Given a data set, the use of a particular statistical learning method is warranted if it results in a low test error.



VALIDATION (HOLD-OUT) SET

- Suppose that we would like to estimate the test error associated with fitting a particular statistical learning method on a set of observations.
- The validation set approach is a very simple strategy validation for this task. It involves randomly dividing the available set of observations into two parts, a training set and a validation set or hold-out set.
- The model is fit on the training set, and the fitted model is used to predict the responses for the observations in the validation set.
- The resulting validation set error rate - typically assessed using MSE in the case of a quantitative response - provides an estimate of the test error rate.

VALIDATION SET MSE



- All ten curves indicate that the model with a quadratic term has a dramatically smaller validation set MSE than the model with only a linear term.
- Furthermore, all ten curves indicate that there is not much benefit in including cubic or higher-order polynomial terms in the model.
- As is shown in the right-hand panel, the validation estimate of the test error rate can be highly variable, depending on precisely which observations are included in the training set and which observations are included in the validation set.

DISADVANTAGE OF VALIDATION SET

- In the validation approach, only a subset of the observations - those that are included in the training set rather than in the validation set - are used to fit the model.
- Since statistical methods tend to perform worse when trained on fewer observations, this suggests that the validation set error rate may tend to overestimate the test error rate for the model fit on the entire data set

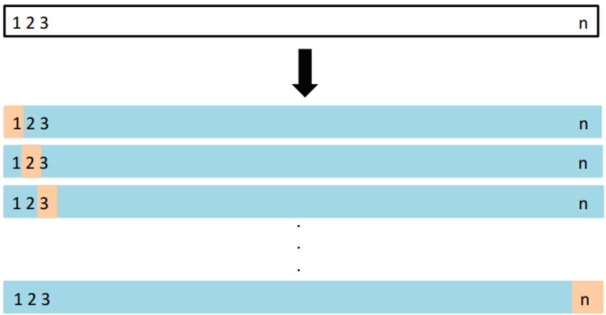
LEAVE ONE OUT CROSS VALIDATION

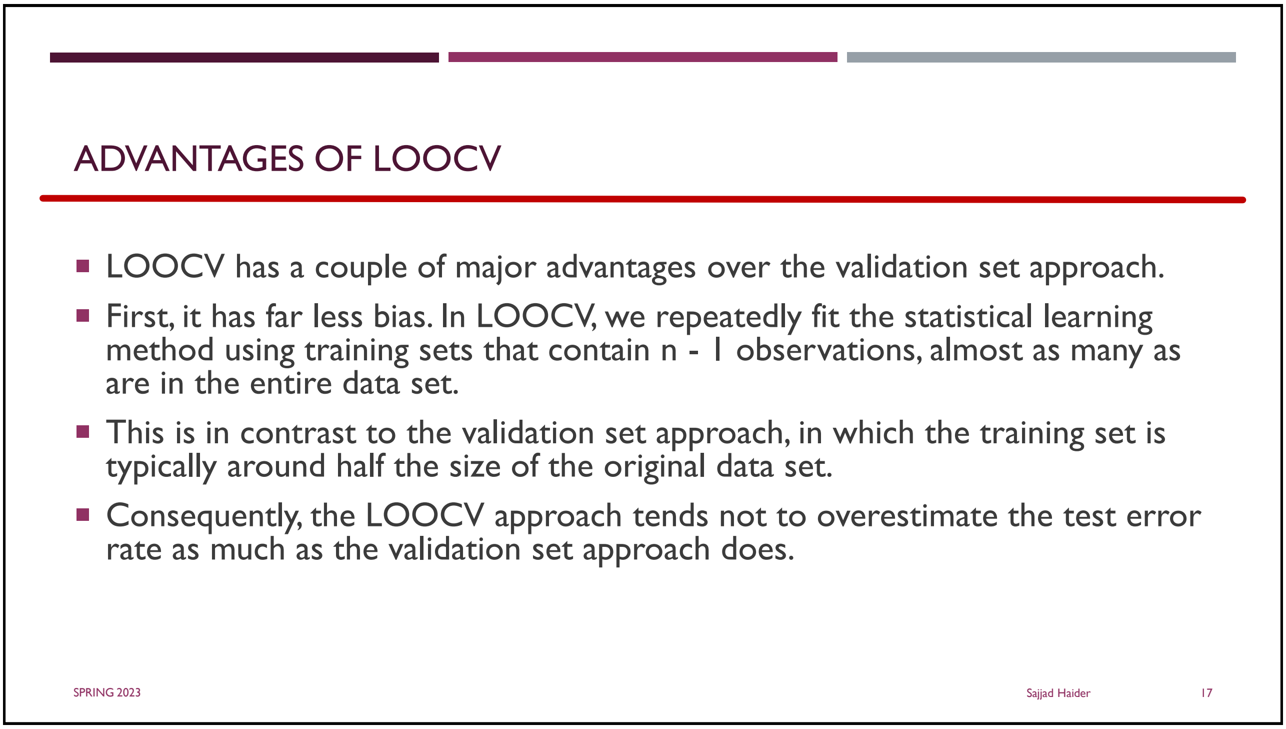
- Like the validation set approach, LOOCV involves splitting the set of observations into two parts.
- However, instead of creating two subsets of comparable size, a single observation (x1, y1) is used for the validation set, and the remaining observations {(x2, y2,…,(xn, yn)} make up the training set.
- The statistical learning method is fit on the n - 1 training observations, and a prediction ^ y1 is made for the excluded observation, using its value x1.
- Since (x1, y1) was not used in the fitting process, MSE1 = (y1 - y^1)2 provides an approximately unbiased estimate for the test error.

LEAVE ONE OUT CROSS VALIDATION (CONT'D)

- We can repeat the procedure by selecting (x2, y2) for the validation data, training the statistical learning procedure on the n - 1 observations {(x1, y1), (x3, y3),…, (xn; yn)}, and computing MSE2 = (y2-y^2)2.
- Repeating this approach n times produces n squared errors, MSE1,…, MSEn.
- The LOOCV estimate for the test MSE is the average of these n test error estimates:

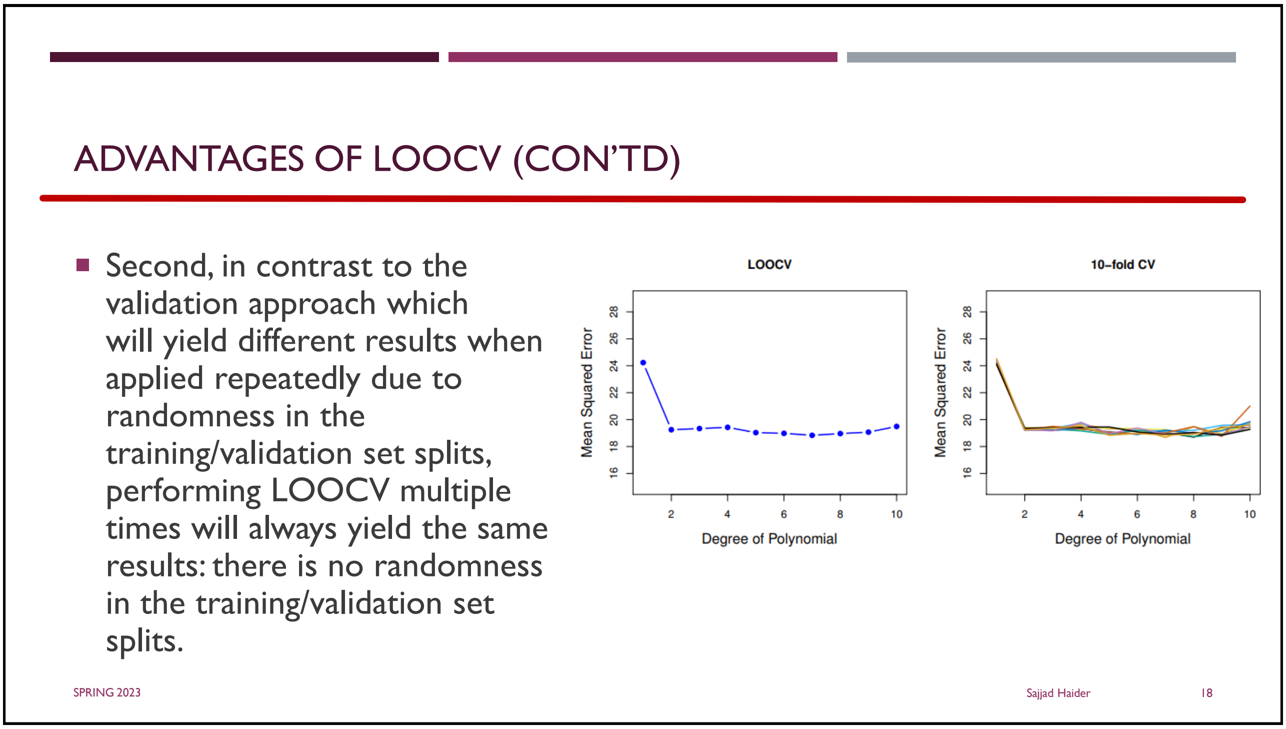
$$\mathrm{CV}_{(n)} = \frac{1}{n}\sum_{i=1}^{n}\mathrm{MSE}_i.$$





ADVANTAGES OF LOOCV

- LOOCV has a couple of major advantages over the validation set approach.
- First, it has far less bias. In LOOCV, we repeatedly fit the statistical learning method using training sets that contain n - 1 observations, almost as many as are in the entire data set.
- This is in contrast to the validation set approach, in which the training set is typically around half the size of the original data set.
- Consequently, the LOOCV approach tends not to overestimate the test error rate as much as the validation set approach does.



ADVANTAGES OF LOOCV (CON'TD)

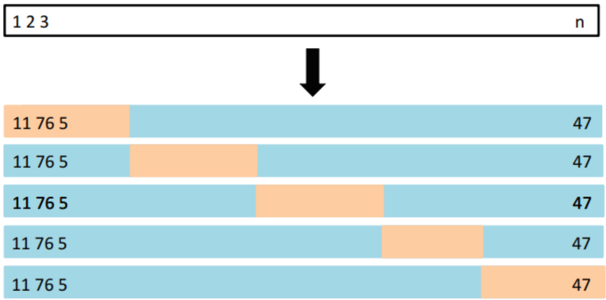
- Second, in contrast to the validation approach which will yield different results when applied repeatedly due to randomness in the training/validation set splits, performing LOOCV multiple times will always yield the same results: there is no randomness in the training/validation set splits.

K-FOLD CROSS VALIDATION

- An alternative to LOOCV is k-fold CV. This approach involves randomly k-fold CV dividing the set of observations into k groups, or folds, of approximately equal size.
- The first fold is treated as a validation set, and the method is fit on the remaining k - 1 folds. The mean squared error, MSE1, is then computed on the observations in the held-out fold.
- This procedure is repeated k times; each time, a different group of observations is treated as a validation set. This process results in k estimates of the test error, MSE1, MSE2, …, MSEk.

$$\mathrm{CV}_{(k)} = \frac{1}{k}\sum_{i=1}^{k} \mathrm{MSE}_i.$$

K-FOLD CROSS VALIDATION (CONT'D)

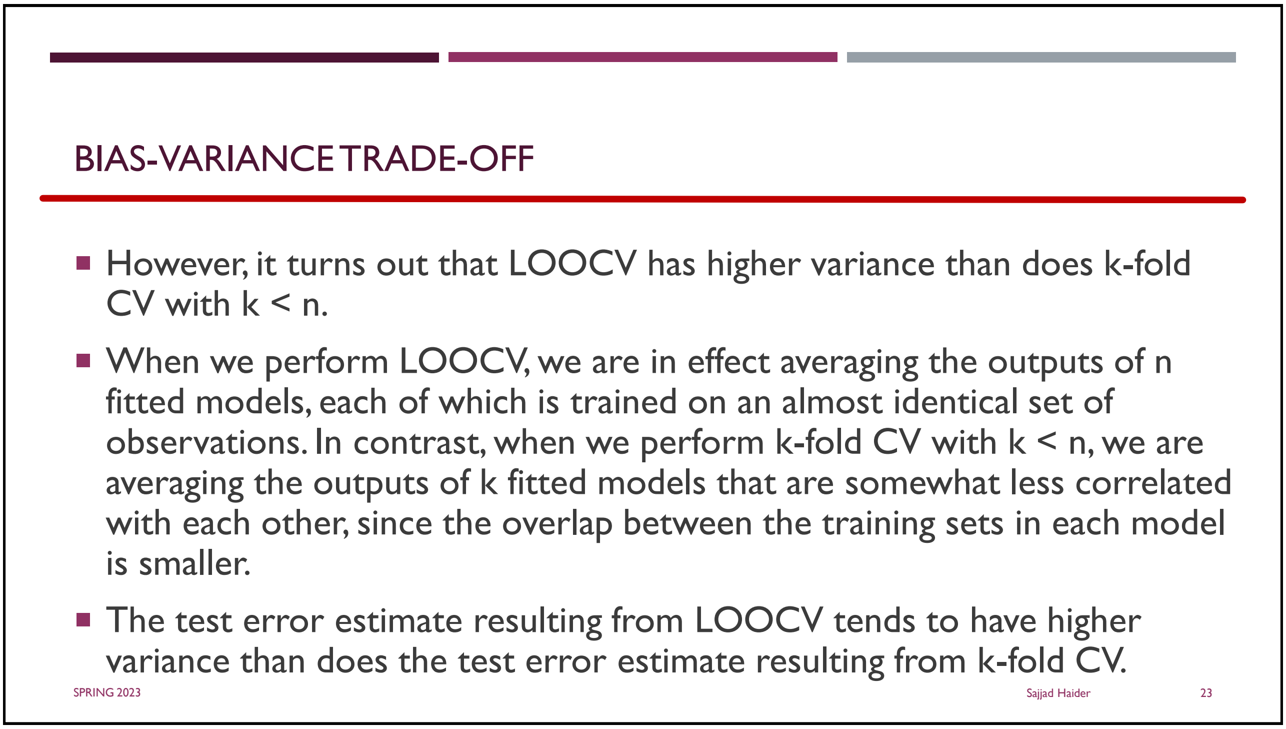


BIAS-VARIANCE TRADE-OFF

- k-fold CV with k < n has a computational advantage to LOOCV.
- But putting computational issues aside, a less obvious but potentially more important advantage of k-fold CV is that it often gives more accurate estimates of the test error rate than does LOOCV. This has to do with a bias-variance trade-off.
- Remember that the validation set approach can lead to overestimates of the test error rate, since in this approach the training set used to fit the statistical learning method contains only half the observations of the entire data set.

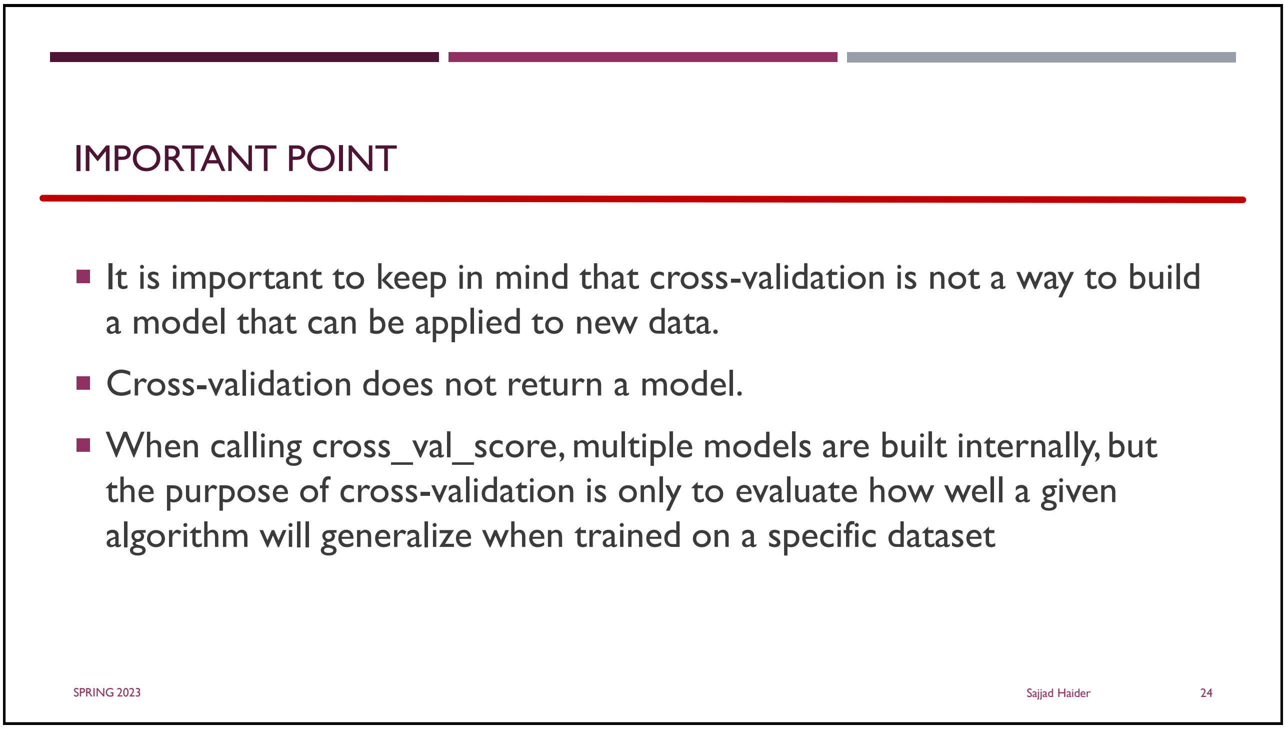
BIAS-VARIANCE TRADE-OFF

- Using this logic, it is not hard to see that LOOCV will give approximately unbiased estimates of the test error, since each training set contains n-1 observations, which is almost as many as the number of observations in the full data set.
- And performing k-fold CV for, say, k = 5 or k = 10 will lead to an intermediate level of bias, since each training set contains approximately (k - 1)n/k observations - fewer than in the LOOCV approach, but substantially more than in the validation set approach.
- Therefore, from the perspective of bias reduction, it is clear that LOOCV is to be preferred to k-fold CV.



BIAS-VARIANCE TRADE-OFF

- However, it turns out that LOOCV has higher variance than does k-fold CV with k < n.
- When we perform LOOCV, we are in effect averaging the outputs of n fitted models, each of which is trained on an almost identical set of observations. In contrast, when we perform k-fold CV with k < n, we are averaging the outputs of k fitted models that are somewhat less correlated with each other, since the overlap between the training sets in each model is smaller.
- The test error estimate resulting from LOOCV tends to have higher variance than does the test error estimate resulting from k-fold CV.



IMPORTANT POINT

- It is important to keep in mind that cross-validation is not a way to build a model that can be applied to new data.
- Cross-validation does not return a model.
- When calling cross_val_score, multiple models are built internally, but the purpose of cross-validation is only to evaluate how well a given algorithm will generalize when trained on a specific dataset

DIMENSION OF DATASETS

- The choice of data representation and selection, reduction, or transformation of features is probably one of the most important issue in machine learning.
- A large number of features can make available samples of data relatively insufficient for model building.
- The three main dimensions of preprocessed data sets are columns (features), rows (cases or samples), and values of the features. Therefore, the three basic operations in a data-reduction process are delete a column, delete a row, and reduce the number of values in a column (smooth a feature).

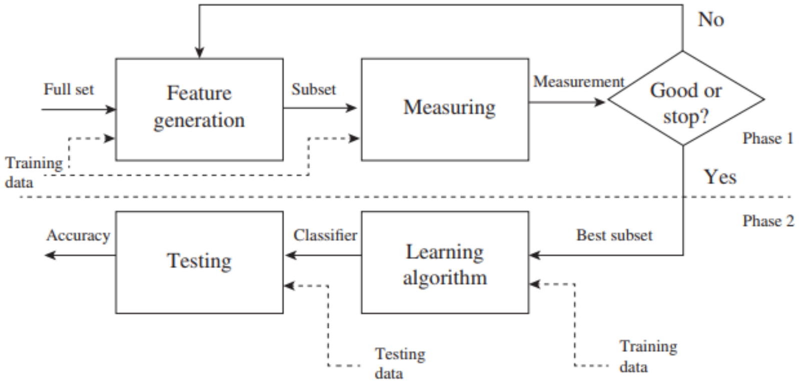
DIMENSION REDUCTION

- There are other operations that reduce dimensions, but the new data are unrecognizable when compared with the original data set
- One approach is the replacement of a set of initial features with a new composite feature.
- For example, if samples in a data set have two features, person's height and person's weight, it is possible for some applications in the medical domain to replace these two features with only one body mass index, which is proportional to the quotient of the initial two features.

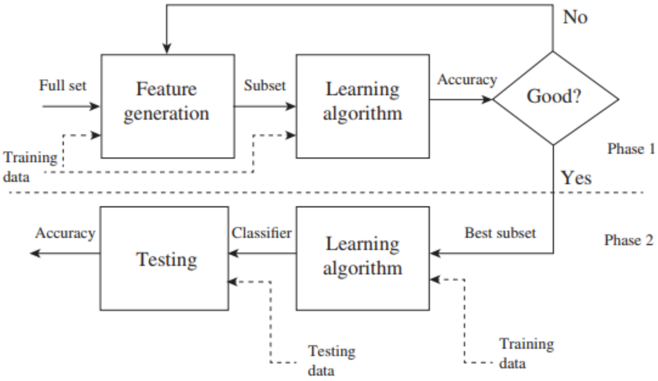
FEATURE SELECTION

- Feature selection models/methods can be further classified as filter and wrapper.
- In the filter model the selection of features is done as a preprocessing activity, without trying to optimize the performance of any specific data-mining technique directly. This is usually achieved through an (ad hoc) evaluation function using a search method in order to select a subset of features that maximizes this function.
- Wrapper methods select features by "wrapping" the search around the selected learning algorithm and evaluate feature subsets based on the learning performance of the data-mining technique for each candidate feature subset.

FILTER



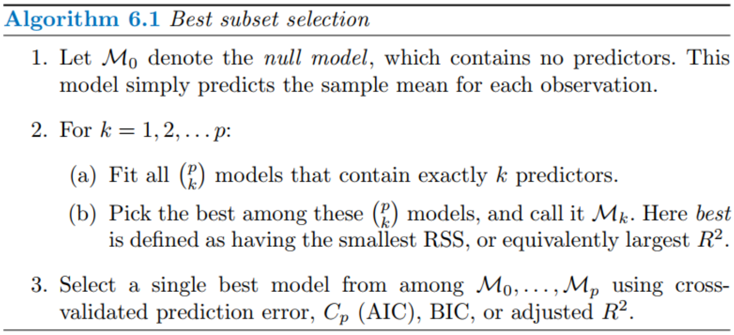
WRAPPER



BEST SUBSET

- To perform best subset selection, we fit a separate least squares regression for each possible combination of the p predictors.
- That is, we fit all p models that contain exactly one predictor, all ${}^{P}C_2$ = p(p-1)/2 models that contain exactly two predictors, and so forth.
- We then look at all of the resulting models, with the goal of identifying the one that is best.
- The problem of selecting the best model from among the 2^p possibilities considered by best subset selection is not trivial.

BEST SUBSET ALGORITHM

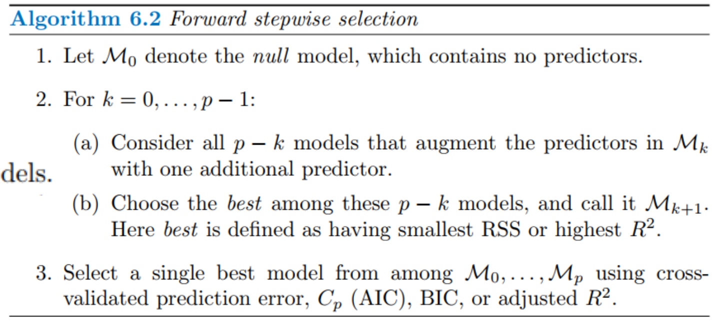


Algorithm 6.1 *Best subset selection*

```
1. Let $\mathcal{M}_0$ denote the null model, which contains no predictors. This
   model simply predicts the sample mean for each observation.
2. For $k = 1, 2, \ldots p$:
   (a) Fit all $\binom{p}{k}$ models that contain exactly $k$ predictors.
   (b) Pick the best among these $\binom{p}{k}$ models, and call it $\mathcal{M}_k$. Here best
       is defined as having the smallest RSS, or equivalently largest $R^2$.
3. Select a single best model from among $\mathcal{M}_0, \ldots, \mathcal{M}_p$ using cross-
   validated prediction error, $C_p$ (AIC), BIC, or adjusted $R^2$.
```

FORWARD STEPWISE SELECTION

- Forward stepwise selection involves fitting one null model, along with p - k models in the kth iteration, for k = 0, …, p - 1.
- This amounts to a total of 1+ $\sum_{k=0}^{p-1}(p-k) = 1+p(p+1)/2$ models.
- This is a substantial difference: when p = 20, best subset selection requires fitting 1,048,576 models, whereas forward stepwise selection requires fitting only 211 models.



Algorithm 6.2 *Forward stepwise selection*

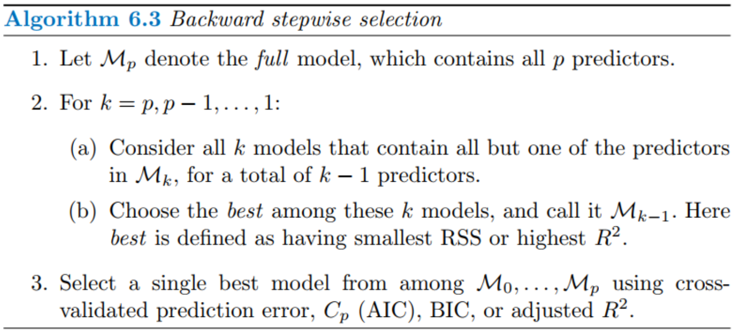
```
1. Let $\mathcal{M}_0$ denote the null model, which contains no predictors.
2. For $k = 0, \ldots, p-1$:
   (a) Consider all $p - k$ models that augment the predictors in $\mathcal{M}_k$
       with one additional predictor.
   (b) Choose the best among these $p - k$ models, and call it $\mathcal{M}_{k+1}$.
       Here best is defined as having smallest RSS or highest $R^2$.
3. Select a single best model from among $\mathcal{M}_0, \ldots, \mathcal{M}_p$ using cross-
   validated prediction error, $C_p$ (AIC), BIC, or adjusted $R^2$.
```

BEST MODEL NOT GUARANTEED IN FORWARD SELECTION

- Though forward stepwise tends to do well in practice, it is not guaranteed to find the best possible model out of all 2p models containing subsets of the p predictors.
- For instance, suppose that in a given data set with p = 3 predictors, the best possible one-variable model contains X1, and the best possible two-variable model instead contains X2 and X3.
- Then forward stepwise selection will fail to select the best possible two-variable model, because M1 will contain X1, so M2 must also contain X1 together with one additional variable.

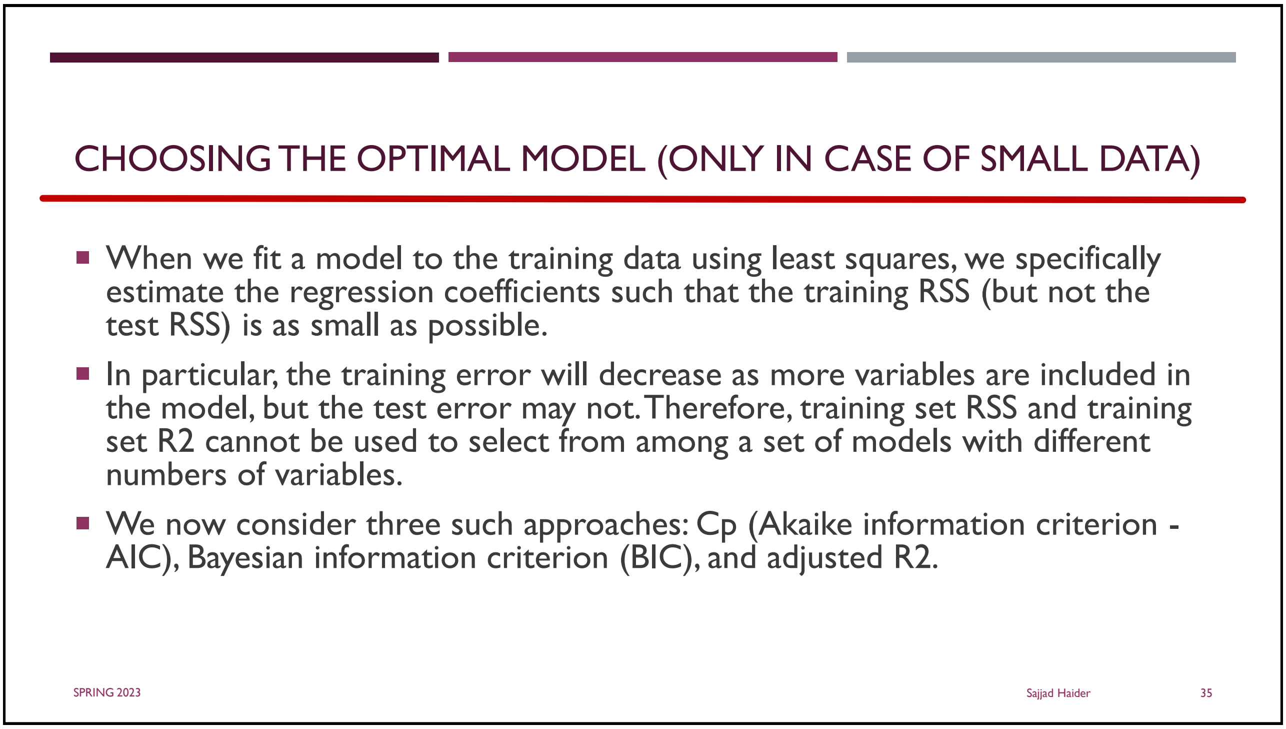
BACKWARD STEPWISE SELECTION

- Unlike forward stepwise selection, backward selection begins with the full least squares model containing all p predictors, and then iteratively removes the least useful predictor, one-at-a-time.



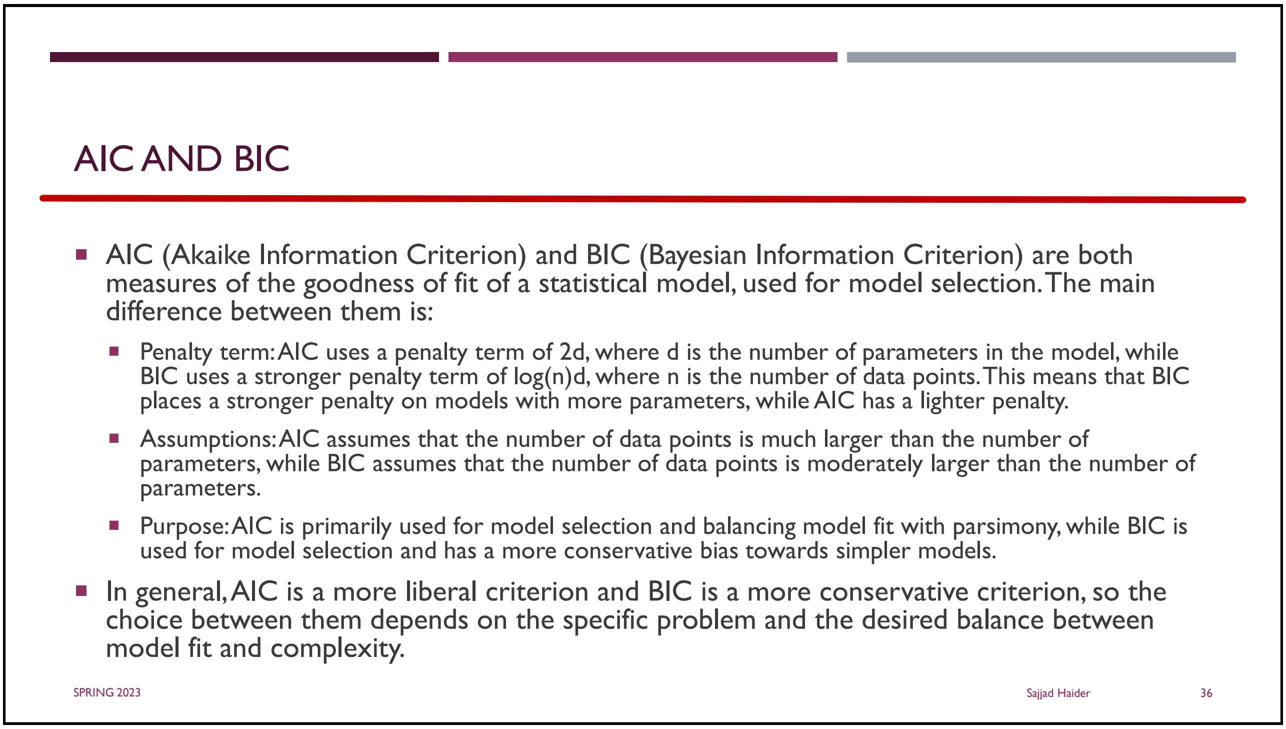
Algorithm 6.3 *Backward stepwise selection*

```
1. Let $\mathcal{M}_p$ denote the *full* model, which contains all $p$ predictors.
2. For $k = p, p-1, \ldots, 1$:
    (a) Consider all $k$ models that contain all but one of the predictors
        in $\mathcal{M}_k$, for a total of $k-1$ predictors.
    (b) Choose the *best* among these $k$ models, and call it $\mathcal{M}_{k-1}$. Here
        *best* is defined as having smallest RSS or highest $R^2$.
3. Select a single best model from among $\mathcal{M}_0, \ldots, \mathcal{M}_p$ using cross-
   validated prediction error, $C_p$ (AIC), BIC, or adjusted $R^2$.
```



CHOOSING THE OPTIMAL MODEL (ONLY IN CASE OF SMALL DATA)

- When we fit a model to the training data using least squares, we specifically estimate the regression coefficients such that the training RSS (but not the test RSS) is as small as possible.
- In particular, the training error will decrease as more variables are included in the model, but the test error may not. Therefore, training set RSS and training set R2 cannot be used to select from among a set of models with different numbers of variables.
- We now consider three such approaches: Cp (Akaike information criterion - AIC), Bayesian information criterion (BIC), and adjusted R2.



AIC AND BIC

- AIC (Akaike Information Criterion) and BIC (Bayesian Information Criterion) are both measures of the goodness of fit of a statistical model, used for model selection. The main difference between them is:
  - Penalty term: AIC uses a penalty term of 2d, where d is the number of parameters in the model, while BIC uses a stronger penalty term of log(n)d, where n is the number of data points. This means that BIC places a stronger penalty on models with more parameters, while AIC has a lighter penalty.
  - Assumptions: AIC assumes that the number of data points is much larger than the number of parameters, while BIC assumes that the number of data points is moderately larger than the number of parameters.
  - Purpose: AIC is primarily used for model selection and balancing model fit with parsimony, while BIC is used for model selection and has a more conservative bias towards simpler models.
- In general, AIC is a more liberal criterion and BIC is a more conservative criterion, so the choice between them depends on the specific problem and the desired balance between model fit and complexity.

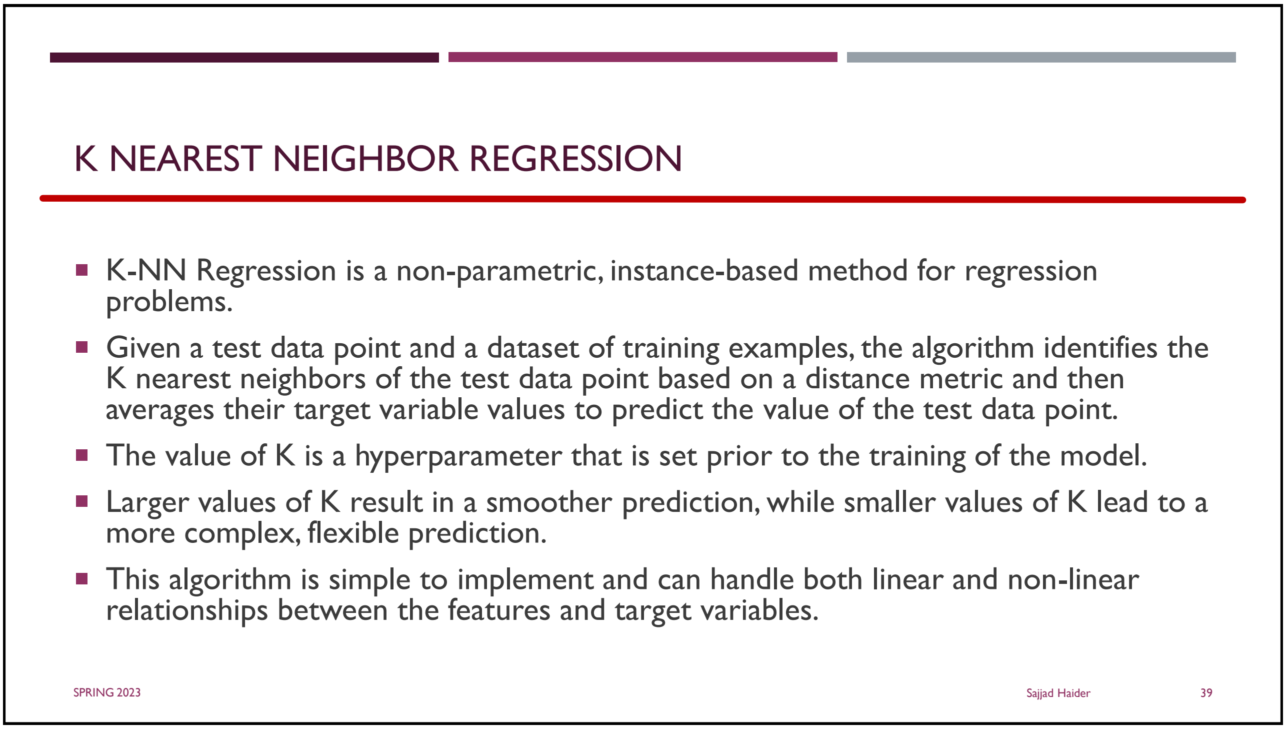
AIC AND BIC (CONT'D)

$$\mathrm{AIC} = \frac{1}{n}\left(\mathrm{RSS} + 2d\hat{\sigma}^2\right) \qquad \mathrm{BIC} = \frac{1}{n}\left(\mathrm{RSS} + \log(n)d\hat{\sigma}^2\right)$$

- where ^ σ2 is an estimate of the variance of the error.
- Notice that BIC replaces the 2dσ^2 used by AIC with a log(n)dσ^2 term, where n is the number of observations.
- Since log n > 2 for any n > 7, the BIC statistic generally places a heavier penalty on models with many variables, and hence results in the selection of smaller models than AIC.

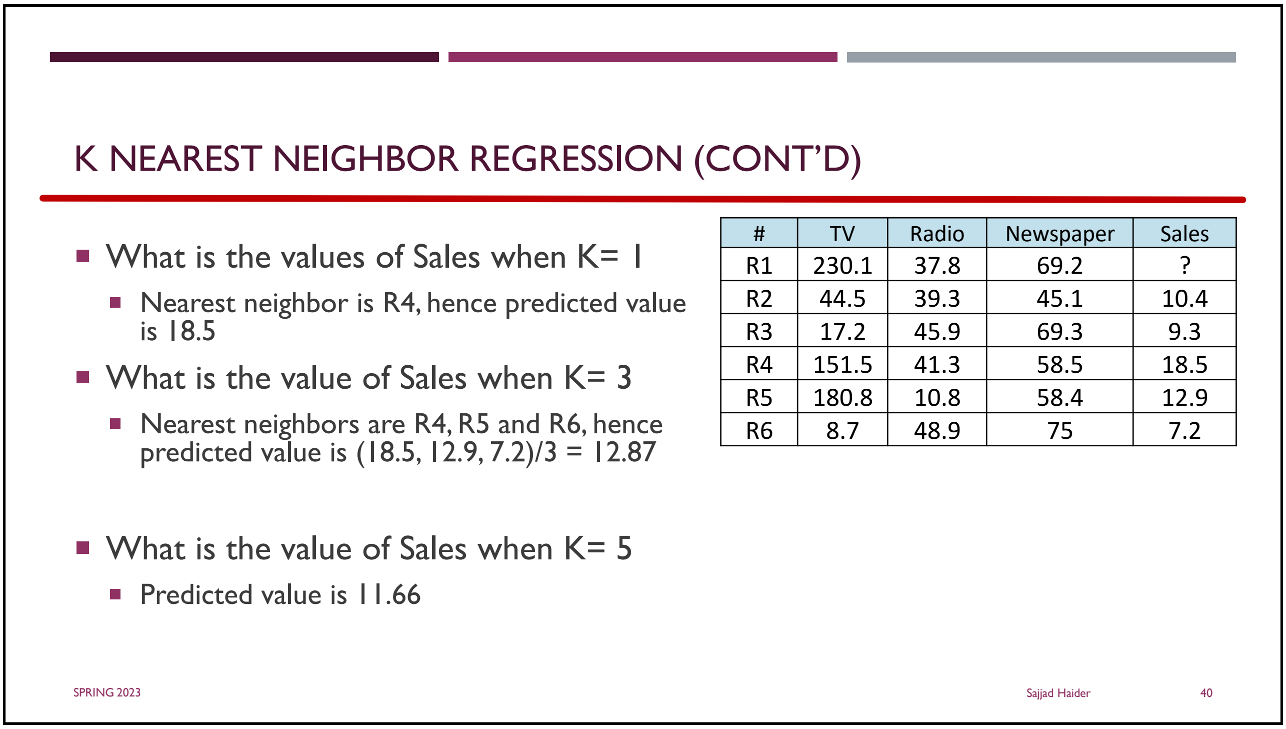
VALIDATION AND CROSS VALIDATION

- In the past, performing cross-validation was computationally prohibitive for many problems with large p and/or large n, and so AIC, BIC, Cp, and adjusted R2 were more attractive approaches for choosing among a set of models.
- However, nowadays with fast computers, the computations required to perform cross-validation are hardly ever an issue. Thus, cross validation is a very attractive approach for selecting from among a number of models under consideration.



K NEAREST NEIGHBOR REGRESSION

- K-NN Regression is a non-parametric, instance-based method for regression problems.
- Given a test data point and a dataset of training examples, the algorithm identifies the K nearest neighbors of the test data point based on a distance metric and then averages their target variable values to predict the value of the test data point.
- The value of K is a hyperparameter that is set prior to the training of the model.
- Larger values of K result in a smoother prediction, while smaller values of K lead to a more complex, flexible prediction.
- This algorithm is simple to implement and can handle both linear and non-linear relationships between the features and target variables.



K NEAREST NEIGHBOR REGRESSION (CONT'D)

- What is the values of Sales when K= 1
  - Nearest neighbor is R4, hence predicted value is 18.5
- What is the value of Sales when K= 3
  - Nearest neighbors are R4, R5 and R6, hence predicted value is (18.5, 12.9, 7.2)/3 = 12.87
- What is the value of Sales when K= 5
  - Predicted value is 11.66

#	TV	Radio	Newspaper	Sales
R1	230.1	37.8	69.2	?
R2	44.5	39.3	45.1	10.4
R3	17.2	45.9	69.3	9.3
R4	151.5	41.3	58.5	18.5
R5	180.8	10.8	58.4	12.9
R6	8.7	48.9	75	7.2