Statistics 452: Statistical Learning and Prediction

Chapter 2: Statistical Learning

Brad McNeney

Statistical Learning

Example 1: Advertising Data

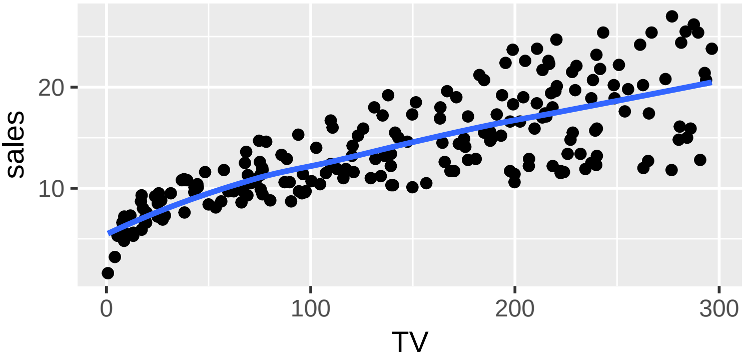
- Sales (in thousands of units), and advertising budgets in thousands of dollars for TV, radio and newspaper for 200 markets.

```
uu <- url("http://faculty.marshall.usc.edu/gareth-james/ISL/Advertising.csv")
advert <- read.csv(uu,row.names=1)
head(advert)
```

```
##      TV radio newspaper sales
## 1 230.1  37.8      69.2  22.1
## 2  44.5  39.3      45.1  10.4
## 3  17.2  45.9      69.3   9.3
## 4 151.5  41.3      58.5  18.5
## 5 180.8  10.8      58.4  12.9
## 6   8.7  48.9      75.0   7.2
```

Relationship Between Sales and TV

```
library(ggplot2)
ggplot(advert,aes(x=TV,y=sales)) +
  geom_point() + geom_smooth(se=FALSE)
```



- The smoother is not constrained to be linear, but is nearly so.
- What sort of return on investment do we get from increasing TV ads?

Exercise

- Do similar scatterplots of `Sales` *vs* `Radio` and `Sales` *vs* `Newspaper`.
  - Try smoothing with an unconstrained smoother (default) and a linear smoother (`geom_smooth(method="lm")`)
  - Which medium provides the best return on investment?

Terminology

- Advertising budgets X_1=TV, X_2=Radio and X_3=Newspaper are **inputs** or **explanatory variables** or **predictors** or **features**
  - Let $X = (X_1, X_2, X_3)$.
- Sales Y is the **output** or **response variable**

Model

- A general model is
$$Y = f(X) + \epsilon$$
where
  - f is a fixed but unknown function that is the **systematic** component of the model
  - ϵ is an error component, assumed to be independent of X and to have mean zero.

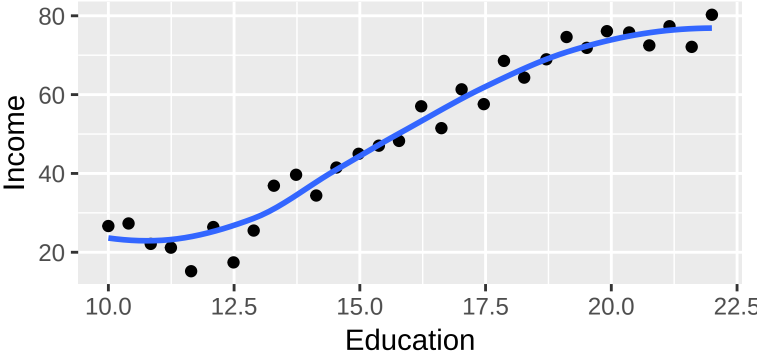
Example 2: Income data

```r
uu <- url("http://faculty.marshall.usc.edu/gareth-james/ISL/Income1.csv")
income <- read.csv(uu,row.names=1)
head(income)
```

```
##   Education   Income
## 1  10.00000 26.65884
## 2  10.40134 27.30644
## 3  10.84281 22.13241
## 4  11.24415 21.16984
## 5  11.64548 15.19263
## 6  12.08696 26.39895
```

Relationship Between Income and Education

```r
ggplot(income,aes(x=Education,y=Income)) +
  geom_point() + geom_smooth(se=FALSE)
```



- Here the relationship is non-linear.
- What is the effect of increasing education?
  - Depends; e.g., not much at low and high education

Statistical Learning

- Approaches for
  - estimating f
  - quantifying the accuracy of the estimate

Why estimate $f(X)$?

- Two main goals:
  1. prediction
  2. inference

Prediction

- Since the errors average to zero, $f(X)$ is a reasonable prediction of a new Y.
- Notation: Let $\hat{f}$ denote an estimate of f and $\hat{Y}$ an estimate of Y.
- Based on $\hat{f}$ the estimate of Y is

$$\hat{Y} = \hat{f}(X)$$

- For prediction, $\hat{f}$ can be a "black box".
  - We do not really care about the details of $\hat{f}$, only that its predictions $\hat{Y}$ are accurate.

Accuracy of $\hat{Y}$

- There are two components
  - reducible error – $\hat{f}$ as an imperfect estimate of f
  - irreducible error – the model includes the pure error component ϵ, which cannot be predicted using X (assumed independent)
- We will study methods for estimating f that try to minimize the reducible error.

Inference

- Or, should our goal be to "open the box" and see what's inside?
  - See first 4:30 of TED talk by Barbara Englehardt: https://www.youtube.com/watch?v=uC3SfnbCXmw
- We may want to understand the relationship between X and Y.
  - If there are many explanatory variables, can we find a few important variables that explain the most variation in the response?
  - What is the nature of relationships: positive/negative, linear/non-linear?

How to estimate $f(X)$

- Methods can be classified as either
  - parametric, or
  - non-parametric
- In either case, we will use **training data** to train our method to estimate f.
- Notation: Let $x_i = (x_{i1}, \ldots, x_{ip})$ denote the observed predictors and y_i the response for the ith of n independent observations.
  - Then the training data are $\{(x_1, y_1), \ldots, (x_n, y_n)\}$

Parametric Methods

- Two steps:
  1. Specify a form for f that depends on a finite number of parameters
  2. Use the training data to estimate the parameters.
- Example:
  1. A linear model $f(X) = \beta_0 + \beta_1 X_1 + \ldots, +\beta_p X_p$.
  2. Use the method of least squares to estimate $\beta_0, \beta_1, \ldots, \beta_p$.

Drawbacks of Parametric Methods

- The true f may not be well-approximated by the functional form we choose for our parametric model.
- We can choose a very flexible parametric family, but if too flexible we may **overfit**; i.e., the fitted model may follow the error terms.

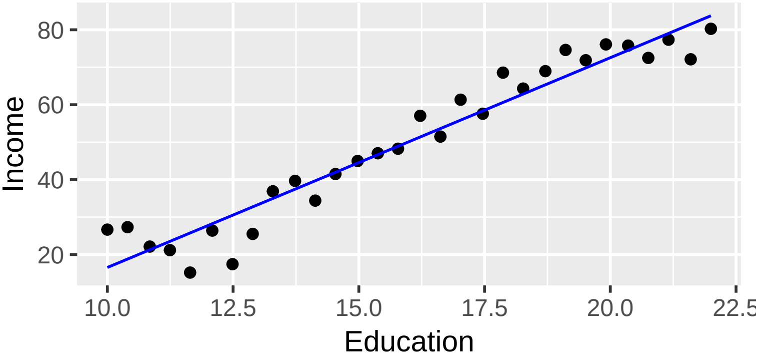
Example: Income data

- Try using powers of `Education` to predict `Income`

```
ifit<- lm(Income ~ Education, data=income)
# grid of Education values
nGrid <- 100
rEd <- with(income,range(Education))
newEd = seq(from=rEd[1],to=rEd[2],length=nGrid)
# Predict income from ifit
newdat <- data.frame(Education = newEd)
pIncome <- predict(ifit,newdata=newdat)
incomePred <- data.frame(Income = pIncome, Education = newEd)
```

Graph the fitted model

```
ggplot(income,aes(x=Education,y=Income)) + geom_point() +
  geom_line(data=incomePred,color="blue")
```



Higher powers

- Repeat for powers of `Education` using `I()`; e.g., for a cubic fit

```
ifit<- lm(Income ~ Education + I(Education^2) + I(Education^3), data=income)
# Now return to code to predict income from ifit and draw fit
```

- At some point, do you get the feeling you are just fitting noise?
    - Fact: If you fit a polynomial of degree 30 you would interpolate the data points.

Non-parametric Methods

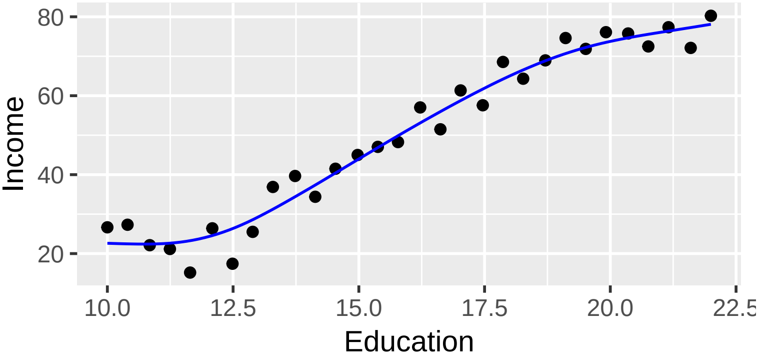
- A model-free specification of the functional form of f.
- Avoid over-fitting by limiting the roughness, or wigglyness of the fitted curve.

Example: Smoothing spline

```r
# install.packages("gam")
library(gam)
sfit <- gam(Income ~ s(Education),data=income)
# Predict income from sfit
pIncome <- predict(sfit,newdata=newdat)
incomePred <- data.frame(Income = pIncome, Education = newEd)
```

Graph the fitted model

```
ggplot(income,aes(x=Education,y=Income)) + geom_point() +
  geom_line(data=incomePred,color="blue")
```

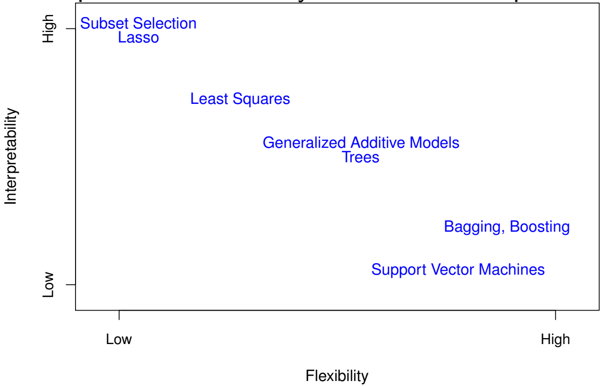


Non-parametric Methods: Drawbacks

- The degree of smoothness was left at its default value – how do we choose this in general?
- Non-parametric methods require more data that a parametric method to train the model to obtain accurate estimates.

Prediction Accuracy *versus* Interpretability

- Figure 2.7 of the text schematically represents the trade-off between prediction accuracy and model interpretability.



- The more flexible the model, the more accurate the predictions, but the less interpretable the model.
  - We will see this by comparing methods as we go.

Supervised *versus* Unsupervised Learning

- When we have measured a response variable the problem is said to be supervised (Chapters 3-9).
- When there is no response, the problem is unsupervised (Chapter 10).
    - We observe x_i; $i = 1, \ldots, n$ and are looking to understand the relationship between the variables, or between the observations (cluster analysis)
    - Cluster analysis is sometimes phrases in terms of looking for a latent (not observed) categorical variable underlying groups in the data.

Regression *versus* Classification

- Regression methods specify models for the conditional mean of the outcome given values of the explanatory variables.
  - Generally, the aim of supervised learning with a quantitative response is regression.
- In classification problems we aim to predict which class an observation belongs to, rather than its mean outcome.
- Some approaches are both; e.g., logistic regression.
  - The outcome may be binary (diseased, not diseased) and we can use a fitted model to classify future observations.
  - But the model fits the mean response given values of the explanatory variables and so is a regression.

Assessing Model Accuracy

Quality of Fit in Regression: MSE

- In regression problems, a popular measure of the quality of a fitted model is the mean squared error (MSE), defined as

$$MSE = \frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{f}(x_i))^2 \tag{1}$$

The Training MSE

- However, we are not especially interested in the MSE from the training data (the training MSE in equation 1).
  - Recall the fact that a high enough polynomial regression can interpolate (see also the wiggly smoothing splines in Figure 2.9 of the text).
  - If all we cared about was training MSE, we'd fit high-degree polynomials.
  - But these would overfit and would give poor predictions of new responses.

The Test MSE

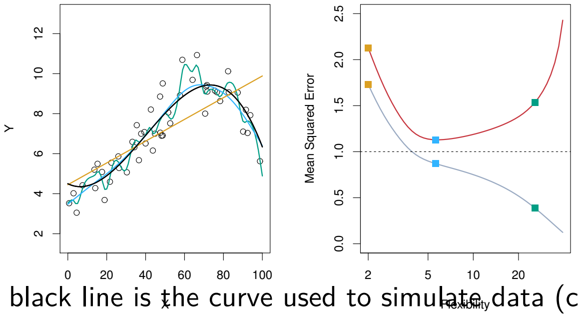
- Instead we are interested in the accuracy of the prediction of new data, called test data. If the training observations $\{(x_1, y_1), \ldots, (x_n, y_n)\}$ are used to produce $\hat{f}$, and we had a large number of test observations (x_0, y_0), the test MSE

$$Ave((y_0 - \hat{f}(x_0))^2)$$

  reflects how well $\hat{f}$ predicts new observations.
- We would like to develop methods that minimize the test MSE.
- **Cross validation** (CV) is a tool to estimate the test MSE.

Training *versus* Test MSE

- Text, simulated data example, Figure 2.9



- The black line is the curve used to simulate data (circles) and the other lines are fitted curves of different flexibility (smoothing splines, Chapter 7).
- In the right panel, the grey line is the training MSE and the red is the test MSE.
  - The "U" shape of the test MSE is typical and reflects the bias-variance trade-off.

Bias-Variance Tradeoff

- For fixed x_0 and y_0, the expected test MSE $E(y_0 - \hat{f}(x_0))^2$, obtained by averaging over repeated estimations of f, can be decomposed as

$$E(y_0 - \hat{f}(x_0))^2 = Var(\hat{f}(x_0)) + [Bias(\hat{f}(x_0))]^2 + Var(\epsilon)$$

  where
  - $Var(\hat{f}(x_0))$ is the variance (spread) of the predictions,
  - $Bias(\hat{f}(x_0))$ is the bias (systematic departure from truth) of the predictions, and
  - $Var(\epsilon)$ is the irreducible error term that is beyond our control
- Generally, the more flexible the method for estimating f the higher the variance and the lower the bias.
  - Initially as we increase flexibility, the variance increase is offset by a decrease in bias, and the test MSE decreases.
  - At some point though the variance increase exceeds the decrease in bias and the expected test MSE increases.

Quality of Fit in Classification

- For categorical Y, the error rate is the proportion of mistaken classifications

$$\frac{1}{n}\sum_{i=1}^{n} I(y_i \neq \hat{y}_i) \tag{2}$$

  where
  - $\hat{y}_i$ is the predicted class label for the ith observation, and
  - $I(y_i \neq \hat{y}_i)$ is an indicator variable that is one if $y_i \neq \hat{y}_i$ and zero if $y_i = \hat{y}_i$.
- Equation (2) is the training error rate. We are more interested in the test error rate:

$$Ave(I(y_0 \neq \hat{y}_0)) \tag{3}$$

  where the average is over new (x_0, y_0).

The Bayes Classifier

- It can be shown that the test error (3) is minimized by the Bayes classifier.
- To a new x_0 the Bayes classifier assigns class label j if $P(Y = j|X = x_0)$ is the largest over all categories j.
- The resulting error rate is called the Bayes error rate – this is a lower bound on the test error rate.
  - This is analogous to the irreducible error from regression.
- We don't know the conditional probabilities $P(Y = j|X = x_0)$ so the Bayes classifier is not practically useful.
  - Suggests we try to estimate the required conditional probabilities. This is the idea behind the K-nearest neighbors classifier (Chapter 4).

Loss Functions

- Reference: Elements of Statistical Learning, Chapter 7.
- We measure the errors between Y and fit $\hat{f}(X)$ by a loss function $L(Y, \hat{f}(X))$.
  - For quantitative Y we mentioned squared error loss

    $$L(Y, \hat{f}(X)) = (Y - \hat{f}(X))^2$$

    which gave us the test MSE.
  - For categorical response, G, we mentioned zero-one loss (misclassification error)

    $$L(Y, \hat{f}(X)) = I(Y \neq \hat{f}(X))$$

    which gave us the test error.
- In general, the test error is the average loss over a test set.