

# Lecture 4: Linear Regression and Classification

STATS 202: Data Mining and Analysis

Linh Tran

tranlm@stanford.edu



Department of Statistics  
Stanford University

June 30, 2021



- ▶ HW1 due next Monday.
  - ▶ Can ask for regrades up to a week from grades being released.
  - ▶ Solutions will be posted on Wednesday.
- ▶ Section on R/Python programming for DS this Friday.
- ▶ Please enroll in Piazza/Gradescope.



- ▶ Regression issues
- ▶ Comparing linear regression to KNN
- ▶ More classification
  - ▶ Logistic regression
  - ▶ Linear/quadratic discriminant analysis



So far, we have:

- ▶ Defined Multiple Linear Regression
- ▶ Discussed how to estimate model parameters
- ▶ Discussed how to test the importance of variables
- ▶ Described one approach to choose a subset of variables
- ▶ Explained how to code dummy indicators

What are some potential issues?



- ▶ Interactions between predictors
- ▶ Non-linear relationships
- ▶ Correlation of error terms
- ▶ Non-constant variance of error (heteroskedasticity)
- ▶ Outliers
- ▶ High leverage points
- ▶ Collinearity
- ▶ Mis-specification

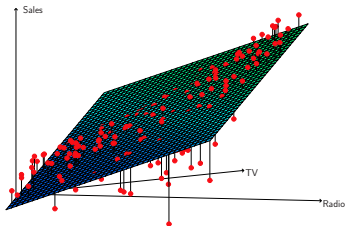


Linear regression has an *additive* assumption, e.g.:

$$\text{sales} = \beta_0 + \beta_1 \cdot \text{tv} + \beta_2 \cdot \text{radio} + \epsilon \quad (1)$$

e.g. An increase of \$ 100 dollars in TV ads correlates to a fixed increase in sales, independent of how much you spend on radio ads.

If we visualize the residuals, it is clear that this is false:





One way to deal with this:

- ▶ Include multiplicative variables (aka interaction variables) in the model

$$sales = \beta_0 + \beta_1 \cdot tv + \beta_2 \cdot radio + \beta_3 \cdot (tv \times radio) + \epsilon \quad (2)$$

- ▶ Makes the effect of TV ads dependent on the radio ads (and vice versa)
- ▶ The *interaction variable* is high when both tv and radio are high



Two ways of including interaction variables (in R):

- ▶ Create a new variable that is the product of the two
- ▶ Specify the interaction in the model formula

```
> lm.fit=lm(Sales~.+Income:Advertising+Price:Age,data=Carseats)
> summary(lm.fit)

Call:
lm(formula = Sales ~ . + Income:Advertising + Price:Age, data =
    Carseats)

Residuals:
    Min       1Q   Median       3Q      Max
-2.921  -0.750   0.018   0.675   3.341

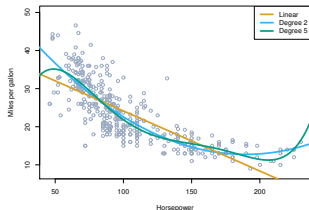
Coefficients:
              Estimate Std. Error t value Pr(>|t|)
(Intercept)   6.575565    1.008747   6.52 2.2e-10 ***
CompPrice     0.092937    0.004118  22.57 < 2e-16 ***
Income        0.010894    0.002604   4.18 3.6e-05 ***
Advertising    0.070246    0.022609   3.11 0.00203 **
Population     0.000159    0.000368   0.43 0.66533
Price       -0.100806    0.007440 -13.55 < 2e-16 ***
ShelveLocGood  4.848676    0.152838  31.72 < 2e-16 ***
ShelveLocMedium 1.953262    0.125768  15.53 < 2e-16 ***
Age          -0.057947    0.015951  -3.63 0.00032 ***
Education    -0.020852    0.019613  -1.06 0.28836
UrbanYes      0.140160    0.112402   1.25 0.21317
USYes        -0.157557    0.148923  -1.06 0.29073
Income:Advertising 0.000751    0.000278   2.70 0.00729 **
Price:Age      0.000107    0.000133   0.80 0.42381
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```





Scatterplots between  $X$  and  $Y$  may reveal non-linear relationships

- **Solution:** Include polynomial terms in the model



$$\begin{aligned} \text{MPG} = & \beta_0 + \beta_1 \cdot \text{horsepower} \\ & + \beta_2 \cdot \text{horsepower}^2 \\ & + \beta_3 \cdot \text{horsepower}^3 + \dots + \epsilon \end{aligned} \quad (3)$$

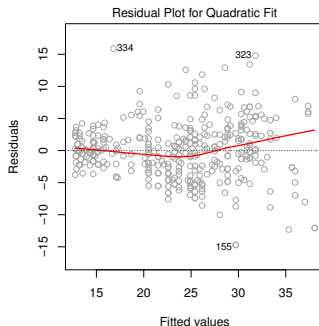
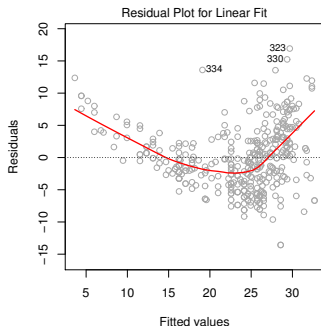


In 2 or 3 dimensions, this is easy to visualize. What do we do when we have too many predictors?



In 2 or 3 dimensions, this is easy to visualize. What do we do when we have too many predictors?

Plot the residuals against the response and look for a pattern:





We assumed that the errors for each sample are independent:

$$y_i = f(x_i) + \epsilon_i : \epsilon_i \stackrel{iid}{\sim} \mathcal{N}(0, \sigma^2) \quad (4)$$



We assumed that the errors for each sample are independent:

$$y_i = f(x_i) + \epsilon_i : \epsilon_i \stackrel{iid}{\sim} \mathcal{N}(0, \sigma^2) \quad (4)$$

When it doesn't hold:

- Invalidates any assertions about Standard Errors, confidence intervals, and hypothesis tests

**Example:** Suppose that by accident, we double the data (i.e. we use each sample twice). Then, the standard errors would be artificially smaller by a factor of  $\sqrt{2}$ .

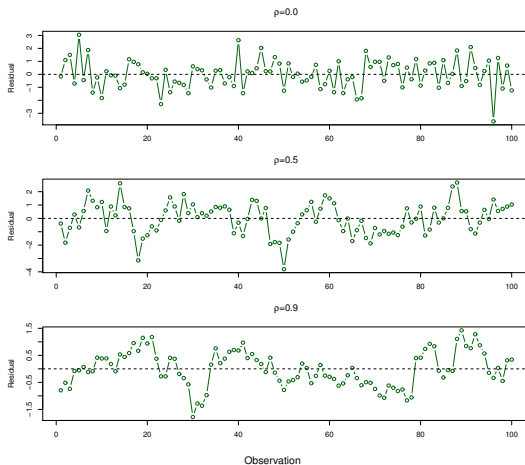


Examples of when this happens:

- ▶ *Time series*: Each sample corresponds to a different point in time. The errors for samples that are close in time are correlated.
- ▶ *Spatial data*: Each sample corresponds to a different location in space.
- ▶ *Clustered data*: Study on predicting height from weight at birth. Suppose some of the subjects in the study are in the same family, their shared environment could make them deviate from  $f(x)$  in similar ways.



Simulations of time series with increasing correlations on  $\epsilon_i$ .

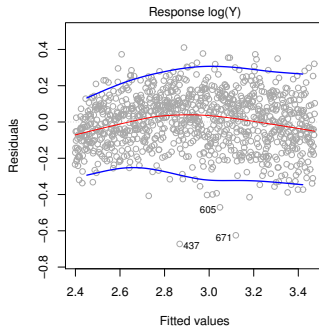
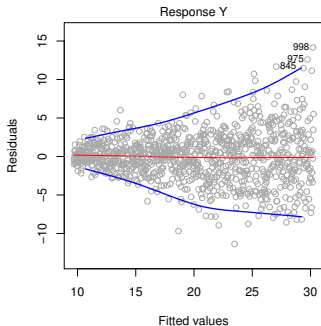


# Non-constant variance of error (heteroskedasticity)



The variance of the error depends on the input value.

To diagnose this, we can plot residuals vs. fitted values:



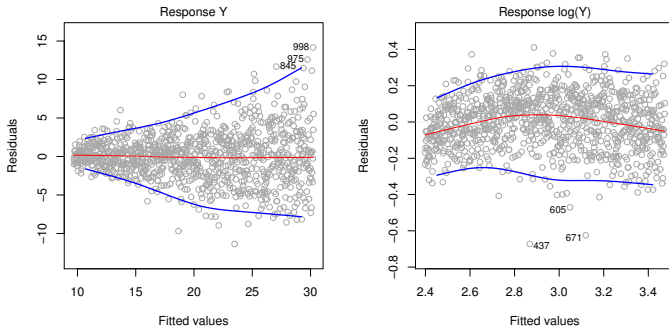


# Non-constant variance of error (heteroskedasticity)



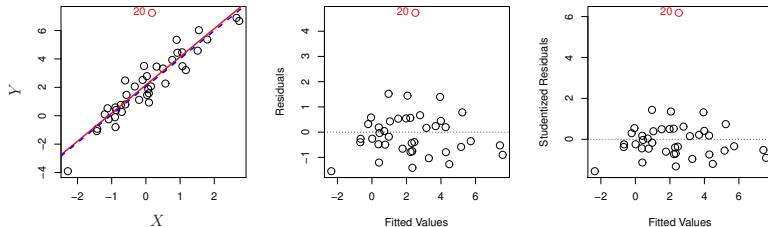
The variance of the error depends on the input value.

To diagnose this, we can plot residuals vs. fitted values:



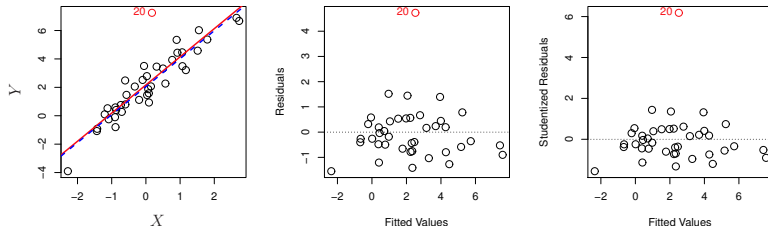
**Solution:** If the trend in variance is relatively simple, we can transform the response using a logarithm, for example.

Outliers are points with very large errors, e.g.



While they may not affect the fit, they might affect our assessment of model quality.

Outliers are points with very large errors, e.g.



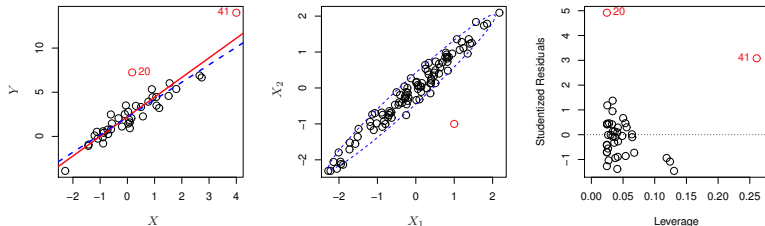
While they may not affect the fit, they might affect our assessment of model quality.

## Possible solutions:

- If we believe an outlier is due to an error in data collection, we can remove it.



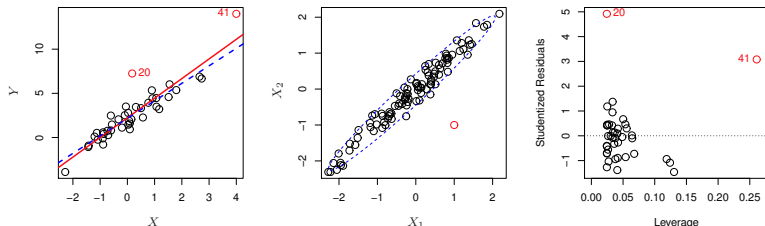
Some samples with extreme inputs have a large effect on  $\hat{\beta}$ .



This can be measured with the *leverage statistic* or *self influence*:

$$h_{ii} = \frac{\partial \hat{y}_i}{\partial y_i} = \underbrace{(\mathbf{X}(\mathbf{X}^\top \mathbf{X})^{-1} \mathbf{X}^\top)}_{\text{Hat matrix}}_{i,i} \in \left[ \frac{1}{n}, 1 \right] \quad (5)$$

Values closer to 1 have high leverage.



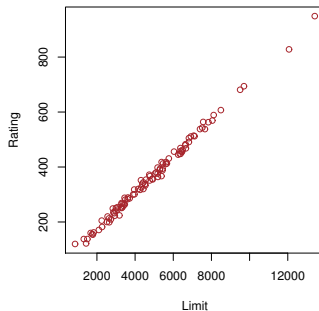
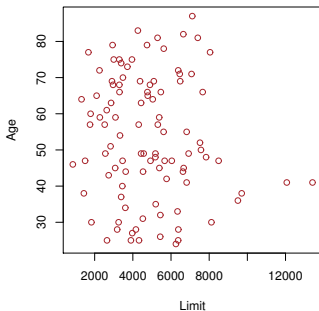
- ▶ The residual  $\hat{\epsilon}_i = y_i - \hat{y}_i$  is an estimate for the noise  $\epsilon_i$
- ▶ The standard error of  $\hat{\epsilon}_i$  is  $\sigma\sqrt{1 - h_{ii}}$
- ▶ A studentized residual is  $\hat{\epsilon}_i$  divided by its standard error
- ▶ It follows a Student- $t$  distribution with  $n - p - 2$  degrees of freedom



Two predictors are collinear if one explains the other well,  
e.g.

$$\text{limit} = a \times \text{rating} + b \quad (6)$$

i.e. they contain the same information





**Problem:** The coefficients become *unidentifiable*.

- ▶ i.e. different coefficients can mean the same fit



**Problem:** The coefficients become *unidentifiable*.

- ▶ i.e. different coefficients can mean the same fit

**Example:** using two identical predictors (*limit*):

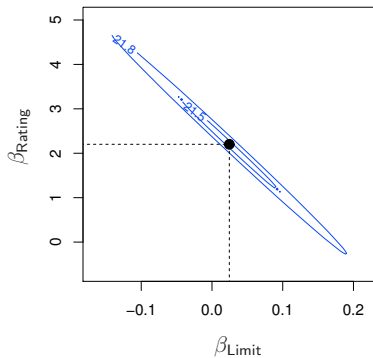
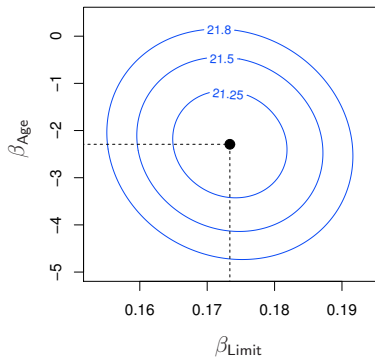
$$balance = \beta_0 + \beta_1 \cdot limit + \beta_2 \cdot limit \quad (7)$$

$$= \beta_0 + (\beta_1 + 100) \cdot limit + (\beta_2 - 100) \cdot limit \quad (8)$$

The fit  $(\hat{\beta}_0, \hat{\beta}_1, \hat{\beta}_2)$  is just as good as  $(\hat{\beta}_0, \hat{\beta}_1 + 100, \hat{\beta}_2 - 100)$



Collinearity results in unstable estimates of  $\beta$ .





If 2 variables are collinear, we can easily diagnose this using their correlation.

A group of  $q$  variables is multilinear if these variables “contain less information” than  $q$  independent variables. Pairwise correlations may not reveal multilinear variables.

The Variance Inflation Factor (VIF) measures how necessary a variable is, or how predictable it is given the other variables:

$$VIF(\hat{\beta}_j) = \frac{1}{1 - R_{X_j|X_{-j}}^2}, \quad (9)$$

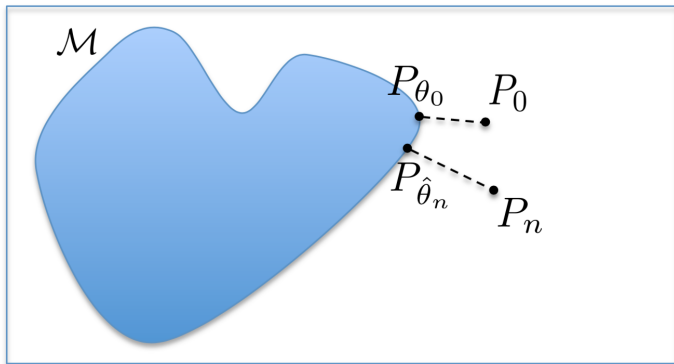
where  $R_{X_j|X_{-j}}^2$  is the  $R^2$  statistic for multiple linear regression of the predictor  $X_j$  onto the remaining predictors.



Three primary ways:

1. Drop one of the correlated features (e.g. Ridge/LASSO).
2. Combine the correlated features (e.g. PCA).
3. More data.

What if our true distribution  $P_0$  **isn't** linear?



Estimates will still converge to a fixed value within our model, e.g.

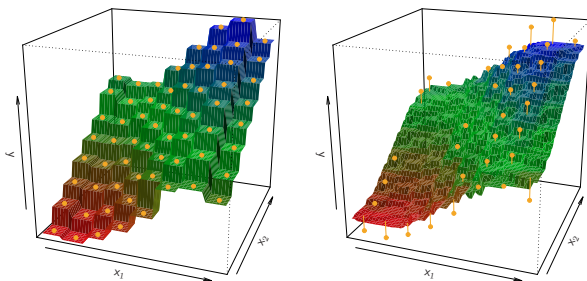
$$\theta_0 \triangleq \arg \min_{\theta} D(P_{\theta}, P_0) : \theta = (\beta_0, \dots, \beta_p) \quad (10)$$



**Linear regression:** prototypical parametric method

**KNN regression:** prototypical nonparametric method

$$\hat{f}_n(x) = \frac{1}{K} \sum_{i \in N_K(x)} y_i \quad (11)$$



Examples of KNN with  $K = 1$  (left) and  $K = 9$  (right)



**Linear regression:** prototypical parametric method

**KNN regression:** prototypical nonparametric method Long story short:

- ▶ KNN is only better when the function  $f_0$  is not linear (and plenty of data)
  - ▶ **Question:** What if the true function  $f_0$  IS linear?



**Linear regression:** prototypical parametric method

**KNN regression:** prototypical nonparametric method Long story short:

- ▶ KNN is only better when the function  $f_0$  is not linear (and plenty of data)
  - ▶ **Question:** What if the true function  $f_0$  IS linear?
- ▶ When  $n$  is not much larger than  $p$ , even if  $f_0$  is nonlinear, linear regression can outperform KNN.



**Linear regression:** prototypical parametric method

**KNN regression:** prototypical nonparametric method Long story short:

- ▶ KNN is only better when the function  $f_0$  is not linear (and plenty of data)
  - ▶ **Question:** What if the true function  $f_0$  IS linear?
- ▶ When  $n$  is not much larger than  $p$ , even if  $f_0$  is nonlinear, linear regression can outperform KNN.
- ▶ KNN has smaller bias, but this comes at a price of (much) higher variance (c.f. overfitting)

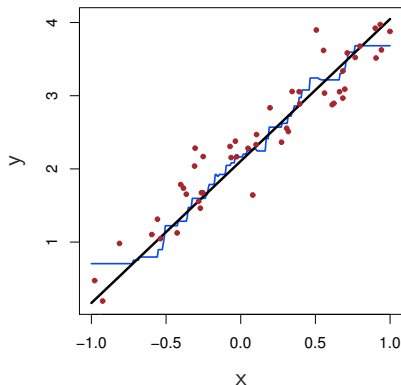
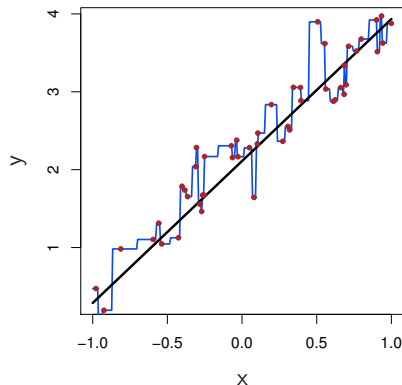


# Comparing Linear Regression to K-nearest neighbors



KNN estimates for a simulation from a linear model

- ▶ True function  $f_0$  is linear

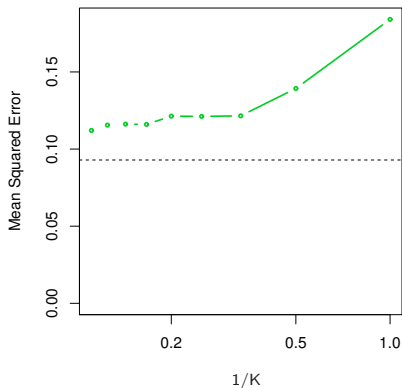
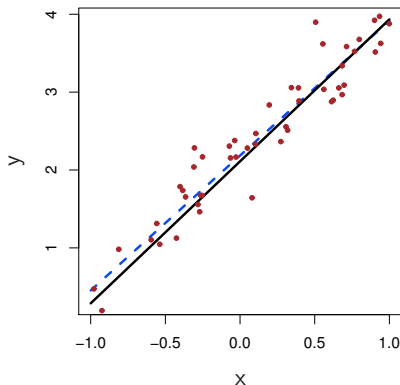


KNN fits with  $K = 1$  (left) and  $K = 9$  (right)



## Linear models dominate KNN

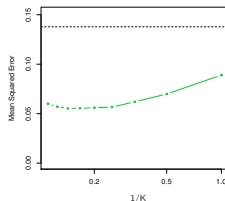
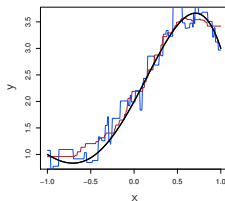
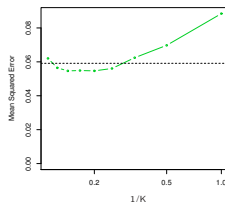
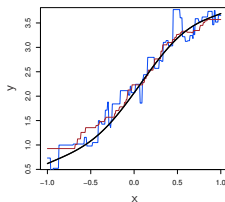
- ▶ We're able to gain statistical efficiency by taking advantage of the linear association



# Comparing Linear Regression to K-nearest neighbors



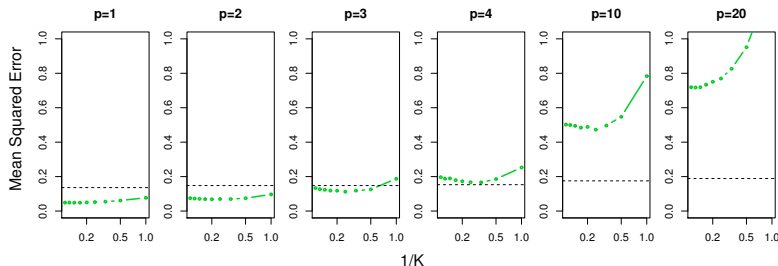
## Increasing deviations from linearity



# Comparing Linear Regression to K-nearest neighbors



When there are more predictors than observations, linear regression dominates



When  $p \gg n$ , each sample has no nearest neighbors, this is known as the curse of dimensionality.

- The variance of KNN regression is very large



**Recall:** Supervised learning with a *qualitative* or *categorical* response. As common (if not more) than regression

- ▶ *Medical diagnosis*: Given the symptoms a patient shows, predict which of 3 conditions they are attributed to
- ▶ *Online banking*: Determine whether a transaction is fraudulent or not, on the basis of the IP address, client's history, etc.
- ▶ *Web searching*: Based on a user's attributes and the string of a web search, predict which link a person will click
- ▶ *Online advertising*: Predict whether a user will click on an ad



**Recall:** In classification, the function  $f_0$  we care about is

$$f_0 \triangleq \mathbb{P}_0[Y = y | X_1, X_2, \dots, X_p] \quad (12)$$

To get a prediction, we use the Bayes Classifier:

$$\hat{y} = \arg \max_y \mathbb{P}_0[Y = y | X_1, X_2, \dots, X_p] \quad (13)$$



**Recall:** In classification, the function  $f_0$  we care about is

$$f_0 \triangleq \mathbb{P}_0[Y = y | X_1, X_2, \dots, X_p] \quad (12)$$

To get a prediction, we use the Bayes Classifier:

$$\hat{y} = \arg \max_y \mathbb{P}_0[Y = y | X_1, X_2, \dots, X_p] \quad (13)$$

**Example:** Suppose  $Y \in \{0, 1\}$ . We could use linear model:

$$\mathbb{P}[Y = 1 | \mathbf{X}] = \beta_0 + \beta_1 X_1 + \dots + \beta_p X_p \quad (14)$$

Problems:

- ▶ This would allow probabilities  $< 0$  and  $> 1$
- ▶ Difficult to extend to more than 2 categories



## An idea:

Let's apply a function to the result to keep it within  $[0, 1]$

$$g^{-1}(z) = \frac{1}{1 + \exp(-z)} \quad (15)$$

i.e.

$$\mathbb{P}[Y = 1|\mathbf{X}] = \frac{1}{1 + \exp(-(\beta_0 + \beta_1 X_1 + \cdots + \beta_p X_p))} \quad (16)$$





## An idea:

Let's apply a function to the result to keep it within  $[0, 1]$

$$g^{-1}(z) = \frac{1}{1 + \exp(-z)} \quad (15)$$

i.e.

$$\mathbb{P}[Y = 1|\mathbf{X}] = \frac{1}{1 + \exp(-(\beta_0 + \beta_1 X_1 + \cdots + \beta_p X_p))} \quad (16)$$

This is equivalent to modeling the log-odds, e.g.

$$\log \left[ \frac{\mathbb{P}[Y = 1|\mathbf{X}]}{\mathbb{P}[Y = 0|\mathbf{X}]} \right] = \beta_0 + \beta_1 X_1 + \cdots + \beta_p X_p \quad (17)$$

n.b.  $\exp(\beta_j)$  is commonly referred to as the *odds-ratio* for  $X_j$



Let  $(\mathbf{x}_1, y_1), (\mathbf{x}_2, y_2), \dots, (\mathbf{x}_n, y_n)$  be our training data.

In the linear model

$$\log \left[ \frac{\mathbb{P}[Y = 1|\mathbf{X}]}{\mathbb{P}[Y = 0|\mathbf{X}]} \right] = \beta_0 + \beta_1 X_1 + \dots + \beta_p X_p, \quad (18)$$

We don't actually observe the left side

- ▶ We observe  $Y \in \{0, 1\}$ , not probabilities
- ▶ This prevents us from using e.g. least squares to estimate our parameters

**Solution:**

Let's try to maximize the probability of our training data



## Solution:

Let's try to maximize the probability of our training data

$$\mathcal{L}(\theta) = \prod_{i=1}^n \mathbb{P}(Y = y_i | \mathbf{X} = \mathbf{x}_i) \quad (19)$$

$$= \prod_{i=1}^n p_i^{y_i} \cdot (1 - p_i)^{1-y_i} \quad (20)$$

where  $p_i = g^{-1}(\beta_0 + \beta_1 x_{1,i} + \cdots + \beta_p x_{p,i})$



## Solution:

Let's try to maximize the probability of our training data

$$\mathcal{L}(\theta) = \prod_{i=1}^n \mathbb{P}(Y = y_i | \mathbf{X} = \mathbf{x}_i) \quad (19)$$

$$= \prod_{i=1}^n p_i^{y_i} \cdot (1 - p_i)^{1-y_i} \quad (20)$$

where  $p_i = g^{-1}(\beta_0 + \beta_1 x_{1,i} + \dots + \beta_p x_{p,i})$

- ▶ We look for  $\theta$  such that  $\mathcal{L}(\theta)$  is maximized
- ▶ aka Maximum likelihood estimation (MLE)
- ▶ Has no closed form solution, so solved with numerical methods (e.g. Newton's method)

**Note:**

We typically deal with the log-likelihood:

$$\ell(\boldsymbol{\theta}) = \log \mathcal{L}(\boldsymbol{\theta}) \quad (21)$$

$$= \sum_{i=1}^n y_i \log(p_i) + (1 - y_i) \log(1 - p_i) \quad (22)$$

$$= \sum_{k=0}^1 \sum_{i=1}^n \mathbb{I}(Y_i = k) \log(\mathbb{P}(k|\mathbf{X} = \mathbf{x}_i)) \quad (23)$$



Given our loss function:

$$\ell(\boldsymbol{\theta}) = \sum_{i=1}^n y_i \log(p_i) + (1 - y_i) \log(1 - p_i) \quad (24)$$



Given our loss function:

$$\ell(\boldsymbol{\theta}) = \sum_{i=1}^n y_i \log(p_i) + (1 - y_i) \log(1 - p_i) \quad (24)$$

Where

$$p_i = \frac{1}{1 + \exp(-Z_i)} \quad (25)$$

$$Z_i = \mathbf{x}_i \boldsymbol{\beta} \quad (26)$$





Given our loss function:

$$\ell(\boldsymbol{\theta}) = \sum_{i=1}^n y_i \log(p_i) + (1 - y_i) \log(1 - p_i) \quad (24)$$

Where

$$p_i = \frac{1}{1 + \exp(-Z_i)} \quad (25)$$

$$Z_i = \mathbf{X}_i \boldsymbol{\beta} \quad (26)$$

We can deriving the gradient using the chain rule:

$$\frac{\partial \ell(\boldsymbol{\theta})}{\partial \boldsymbol{\beta}} = \frac{\partial \ell(\boldsymbol{\theta})}{\partial p_i} \times \frac{\partial p_i}{\partial Z_i} \times \frac{\partial Z_i}{\partial \boldsymbol{\beta}} \quad (27)$$



## Estimating uncertainty

- ▶ We can estimate the Standard Error of each coefficient (e.g. using Fisher's information)

$$\mathbf{I}_{\mathbf{Y}}(\beta) = -\mathbb{E}_{\beta}[\nabla^2 \ell(\beta)] \quad (28)$$

- ▶ The  $z$ -statistic (for logistic regression) is the equivalent of the  $t$ -statistic (in linear regression):

$$z = \frac{\hat{\beta}_j}{\hat{SE}(\hat{\beta}_j)} \quad (29)$$

- ▶ The  $p$ -values are test of the null hypothesis  $\beta_j = 0$



## Example fit

```
> glm.fit=glm(Direction~Lag1+Lag2+Lag3+Lag4+Lag5+Volume ,  
  data=Smarket ,family=binomial)  
> summary(glm.fit)
```

Call:

```
glm(formula = Direction ~ Lag1 + Lag2 + Lag3 + Lag4 + Lag5  
  + Volume, family = binomial, data = Smarket)
```

Deviance Residuals:

Min	1Q	Median	3Q	Max
-1.45	-1.20	1.07	1.15	1.33

Coefficients:

	Estimate	Std. Error	z value	Pr(> z )
(Intercept)	-0.12600	0.24074	-0.52	0.60
Lag1	-0.07307	0.05017	-1.46	0.15
Lag2	-0.04230	0.05009	-0.84	0.40
Lag3	0.01109	0.04994	0.22	0.82
Lag4	0.00936	0.04997	0.19	0.85
Lag5	0.01031	0.04951	0.21	0.83
Volume	0.13544	0.15836	0.86	0.39



Predictors:

- ▶ `student`: 1 if student, 0 otherwise
- ▶ `balance`: credit card balance
- ▶ `income`: person's income

In this dataset there is *confounding*, but little collinearity

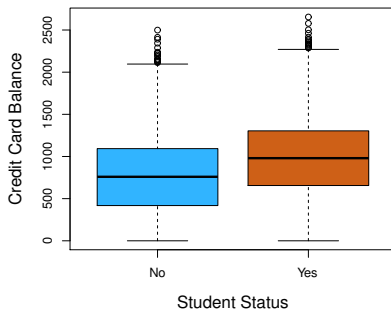
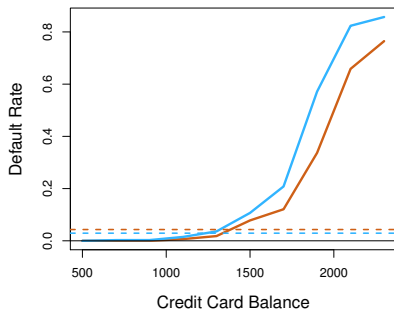
- ▶ Students tend to have higher balances. So, `balance` is explained by `student`, but not very well.
- ▶ People with a high balance are more likely to default.
- ▶ Among people with a given balance, students are less likely to default.

# Example: Predicting credit card default



Predictors:

- ▶ student: 1 if student, 0 otherwise
- ▶ balance: credit card balance
- ▶ income: person's income





Logistic regression using only balance:

	Coefficient	Std. error	Z-statistic	P-value
Intercept	-10.6513	0.3612	-29.5	<0.0001
balance	0.0055	0.0002	24.9	<0.0001

Logistic regression using only student:

	Coefficient	Std. error	Z-statistic	P-value
Intercept	-3.5041	0.0707	-49.55	<0.0001
student[Yes]	0.4049	0.1150	3.52	0.0004

Logistic regression using all 3 predictors:

	Coefficient	Std. error	Z-statistic	P-value
Intercept	-10.8690	0.4923	-22.08	<0.0001
balance	0.0057	0.0002	24.74	<0.0001
income	0.0030	0.0082	0.37	0.7115
student[Yes]	-0.6468	0.2362	-2.74	0.0062



- ▶ The coefficients become unstable when there is collinearity
  - ▶ This also affects the convergence of the fitting algorithm
- ▶ When the classes are well separated, the coefficients become unstable
  - ▶ This is always the case when  $p \geq n - 1$ .
- ▶ Sometimes may not converge
  - ▶ e.g. Needs more iterations



A linear model (like logistic regression). Unlike logistic regression:

- ▶ Does not become unstable when classes are well separated
- ▶ With small  $n$  and  $\mathbf{X}$  approximately normal, is stable
- ▶ Popular when we have  $> 2$  classes





A linear model (like logistic regression). Unlike logistic regression:

- ▶ Does not become unstable when classes are well separated
- ▶ With small  $n$  and  $\mathbf{X}$  approximately normal, is stable
- ▶ Popular when we have  $> 2$  classes

## High level idea:

Model distribution of  $\mathbf{X}$  given  $Y$ , and apply Bayes' theorem, i.e.

$$\mathbb{P}(Y = k | \mathbf{X} = \mathbf{x}) = \frac{\pi_k f_k(\mathbf{x})}{\sum_{l=1}^K \pi_l f_l(\mathbf{x})} \quad (30)$$

- ▶ A common assumption is  $f_k(\mathbf{x})$  is Gaussian



**Example:**  $K = 2$  with Gaussian  $f_k(\mathbf{x})$  and common  $\sigma^2$

$$\mathbb{P}(Y = k | \mathbf{X} = \mathbf{x}) = \frac{\pi_k f_k(\mathbf{x})}{\sum_{l=1}^K \pi_l f_l(\mathbf{x})} \quad (31)$$

$$= \frac{\pi_k \frac{1}{\sqrt{2\pi}\sigma} \exp\left(-\frac{1}{2\sigma^2}(x - \mu_k)^2\right)}{\sum_{l=1}^2 \pi_l \frac{1}{\sqrt{2\pi}\sigma} \exp\left(-\frac{1}{2\sigma^2}(x - \mu_l)^2\right)} \quad (32)$$



**Example:**  $K = 2$  with Gaussian  $f_k(\mathbf{x})$  and common  $\sigma^2$

$$\mathbb{P}(Y = k | \mathbf{X} = \mathbf{x}) = \frac{\pi_k f_k(\mathbf{x})}{\sum_{l=1}^K \pi_l f_l(\mathbf{x})} \quad (31)$$

$$= \frac{\pi_k \frac{1}{\sqrt{2\pi}\sigma} \exp\left(-\frac{1}{2\sigma^2}(x - \mu_k)^2\right)}{\sum_{l=1}^2 \pi_l \frac{1}{\sqrt{2\pi}\sigma} \exp\left(-\frac{1}{2\sigma^2}(x - \mu_l)^2\right)} \quad (32)$$

Taking the log and rearranging gives:

$$\delta_k(x) = x \cdot \frac{\mu_k}{\sigma^2} - \frac{\mu_k^2}{2\sigma^2} + \log(\pi_k) \quad (33)$$



**Example:**  $K = 2$  with Gaussian  $f_k(\mathbf{x})$  and common  $\sigma^2$

$$\mathbb{P}(Y = k | \mathbf{X} = \mathbf{x}) = \frac{\pi_k f_k(\mathbf{x})}{\sum_{l=1}^K \pi_l f_l(\mathbf{x})} \quad (31)$$

$$= \frac{\pi_k \frac{1}{\sqrt{2\pi}\sigma} \exp\left(-\frac{1}{2\sigma^2}(x - \mu_k)^2\right)}{\sum_{l=1}^2 \pi_l \frac{1}{\sqrt{2\pi}\sigma} \exp\left(-\frac{1}{2\sigma^2}(x - \mu_l)^2\right)} \quad (32)$$

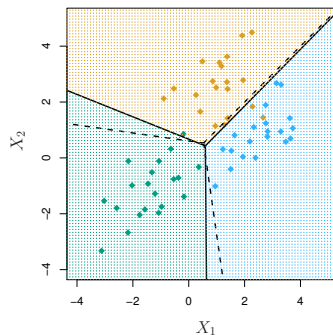
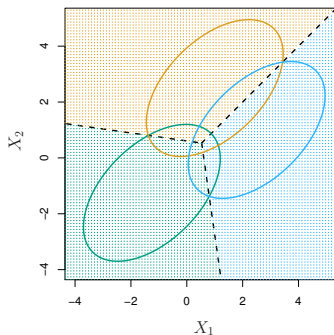
Taking the log and rearranging gives:

$$\delta_k(x) = x \cdot \frac{\mu_k}{\sigma^2} - \frac{\mu_k^2}{2\sigma^2} + \log(\pi_k) \quad (33)$$

If  $\pi_1 = \pi_2$ , our Bayes Classifier is:

$$2x(\mu_1 - \mu_2) > \mu_1^2 - \mu_2^2 \quad (34)$$

## Example of LDA





Similar to LDA

- ▶ Assumes Gaussian  $f_k(\mathbf{x})$
- ▶ Unlike LDA:
  - ▶ Assumes each class has its own covariance matrix ( $\Sigma_k$ )



Similar to LDA

- ▶ Assumes Gaussian  $f_k(\mathbf{x})$
- ▶ Unlike LDA:
  - ▶ Assumes each class has its own covariance matrix ( $\Sigma_k$ )

This results in a quadratic discriminant function:

$$\begin{aligned}\delta_k(x) &= -\frac{1}{2}(x - \mu_k)^\top \Sigma_k^{-1}(x - \mu_k) - \frac{1}{2} \log |\Sigma_k| + \log \pi_k \\ &= -\frac{1}{2}x^\top \Sigma_k^{-1}x + x^\top \Sigma_k^{-1}\mu_k - \frac{1}{2}\mu_k^\top \Sigma_k^{-1}\mu_k - \frac{1}{2} \log |\Sigma_k| + \log \pi_k\end{aligned}\tag{35}$$



Similar to LDA

- ▶ Assumes Gaussian  $f_k(\mathbf{x})$
- ▶ Unlike LDA:
  - ▶ Assumes each class has its own covariance matrix ( $\Sigma_k$ )

This results in a quadratic discriminant function:

$$\begin{aligned}\delta_k(x) &= -\frac{1}{2}(x - \mu_k)^\top \Sigma_k^{-1}(x - \mu_k) - \frac{1}{2} \log |\Sigma_k| + \log \pi_k \\ &= -\frac{1}{2}x^\top \Sigma_k^{-1}x + x^\top \Sigma_k^{-1}\mu_k - \frac{1}{2}\mu_k^\top \Sigma_k^{-1}\mu_k - \frac{1}{2} \log |\Sigma_k| + \log \pi_k\end{aligned}\tag{35}$$

This results in more parameters to fit:

- ▶ LDA:  $Kp$  parameters
- ▶ QDA:  $Kp(p+1)/2$  parameters





Assume a two-class setting with one predictor

## Linear Discriminant Analysis:

$$\log \left[ \frac{p_1(x)}{1 - p_1(x)} \right] = c_0 + c_1 x \quad (36)$$

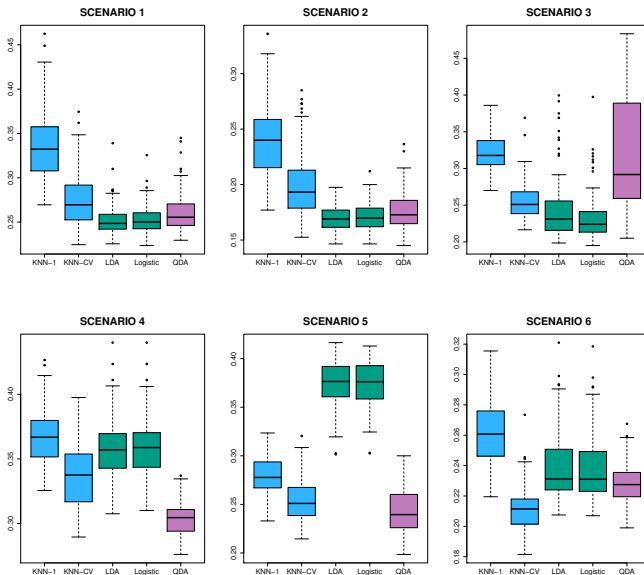
- $c_0$  and  $c_1$  computed using  $\hat{\mu}_0$ ,  $\hat{\mu}_1$ , and  $\hat{\sigma}^2$

## Logistic regression:

$$\log \left[ \frac{\mathbb{P}[Y = 1|x]}{1 - \mathbb{P}[Y = 1|x]} \right] = \beta_0 + \beta_1 x \quad (37)$$

- $\beta_0$  and  $\beta_1$  estimated using MLE

# Comparison of classification methods





[1] ISL. Chapters 3-4.

[2] ESL. Chapters 3.