Introduction to Data Science

ML Analysis and Linear Regression

Gordon Anderson

Linear Regression

- In these notes, we'll discuss the concepts behind linear regression.
- First, an overview of what machine learning is and how we go about performing an ML analysis.
- Then the basics of linear regression, including multiple linear regression.

Machine Learning- ML

- ML is the process of modeling data with an algorithm that learns a set of parameters that describe a generalization of trends or patterns in the data.
- Statistical modeling vs. Machine Learning- what’s the difference?
  - Not much- ML model is often used by another algorithm, such as a recommendation system, to make decisions.
- ML algorithms need training data from which they can learn, and test data which can be used to evaluate the model’s performance on data it has not yet seen.

Machine Learning- ML

- Basic types of ML:
  - **Supervised:** The input to the algorithm are data of the form: $\langle y_i, \vec{X}_i \rangle$, where each *y* represents an outcome, and *X* represents a vector of predictors. This pair is a training example for the ML algorithm. The algorithm learns to predict the outcome, *y*, from the predictors, *X*. The subscript, *i*, indicates the i^{th} row in the data set. Example: linear regression.
  - **Unsupervised:** The algorithm does not receive any examples of outcomes, it learns from data only. Example: K-means clustering.
  - **Semi-supervised:** some of the training data includes outcomes, some does not. We won't be working with this type of ML.

ML analysis- supervised learning:

- Once the data has been obtained and the ML algorithm have been selected:
- Create a training data set, which is a subset of the original data set.
- This can be done manually, by selecting a specific number of rows from the original data set, or by randomly selecting a number of rows for the training set.
- The remainder of the original data is used for testing.

ML Modeling Data

Entire data set:

Outcome

Predictors

kid_score	mom_hs	mom_iq	mom_work	mom_age
65	1	121.1175286	4	27
98	1	89.36188171	4	25
85	1	115.4431649	4	27
83	1	99.44963944	3	25
115	1	92.74571	4	27
98	0	107.9018378	1	18
69	1	138.8931061	4	20
106	1	125.1451195	3	23
102	1	81.61952618	1	24
95	1	95.07306862	1	19
91	1	88.57699772	1	23
58	1	94.85970819	4	24
84	1	88.96280085	4	27
78	1	114.114297	4	26
102	0	100.5340719	2	24
110	1	120.4191456	1	26

Often, the test and train sets are selected at random.

Training set:
<outcome, predictors>

65	1	121.1175286	4	27
98	1	89.36188171	4	25
85	1	115.4431649	4	27
83	1	99.44963944	3	25
115	1	92.74571	4	27
98	0	107.9018378	1	18
69	1	138.8931061	4	20
106	1	125.1451195	3	23

“True” outcomes:

102
95
91
58
84
78
102
110

Testing set:
<predictors>, model outputs predictions

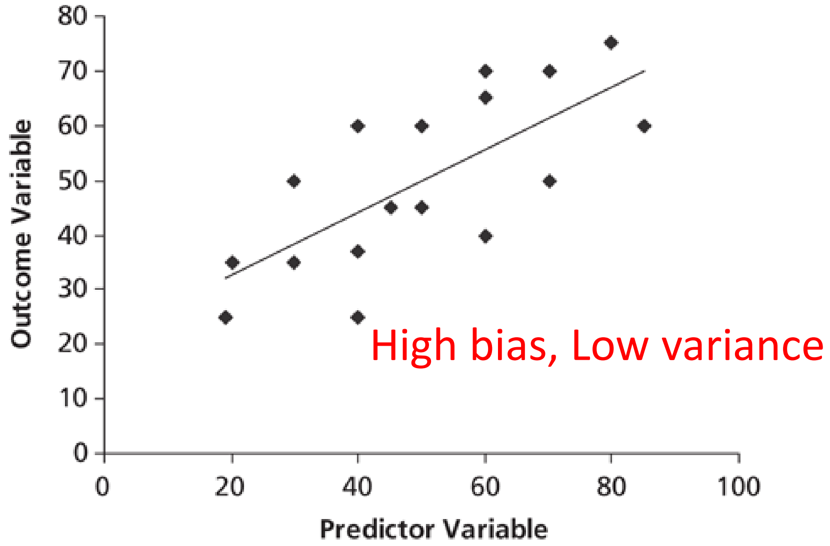
1	81.61952618	1	24
1	95.07306862	1	19
1	88.57699772	1	23
1	94.85970819	4	24
1	88.96280085	4	27
1	114.114297	4	26
0	100.5340719	2	24
1	120.4191456	1	26

ML analysis- model fitting:

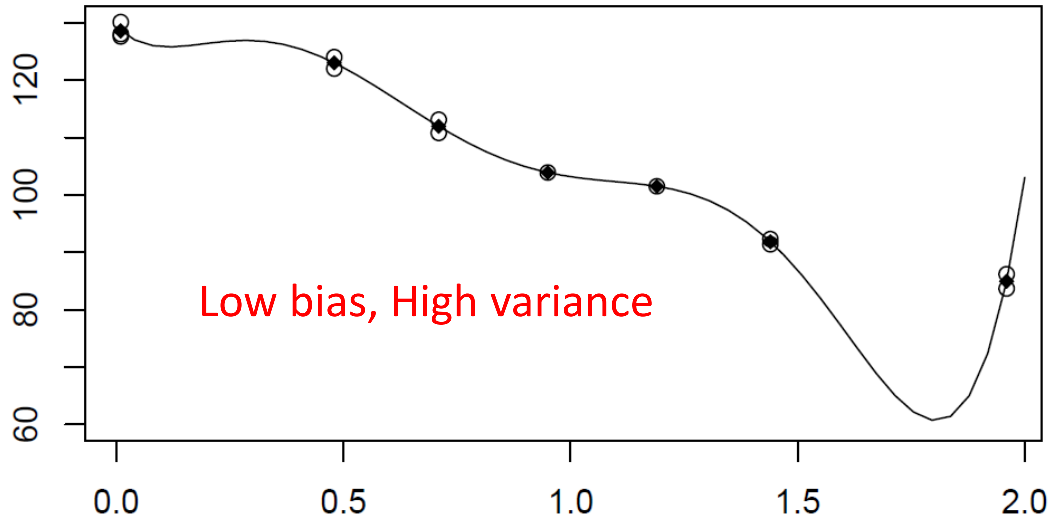
- Next, apply the algorithm to the training data. This is called “fitting” a model.
- Evaluate the quality of the fit. There will be errors, called *residuals*, because we want to learn the general trend in the data.
- The data contains noise. A model with very low fit errors is probably “overfitting” the data- it is not generalizing but fitting the noise. It will not predict well on the test data.
- If the model fit errors are very high, the model is too general and probably did not capture the trend in the data.
- Terms:
  - residuals- difference between estimated model and data (observed).
  - errors- difference between true model and data (unobserved)

Model Fit- bias vs. variance

Modeling is a trade-off between fitting the training set well and generalizing enough to predict new data well.



Error in the fit, but hopefully it generalized well. That is determined when we test its predictions.



A perfect fit, but no generalization:
This model has ***overfit*** these data and
Is most likely a poor predictor.

ML analysis- model evaluation:

- The fitted model can be used on new data to make predictions.
- The fitted model's output is a set of predicted outcomes, $\hat{Y}$, or "Y-hat". The capital letter indicates a vector of values. The "hat" signifies that the values are estimates.
- The length of Y-hat is the same as the number of rows in the test set.
- To evaluate the model's performance, the true outcomes from the test set are compared against the model's predictions.

ML Modeling Steps

Training set:
<outcome, predictors>

65	1	121.1175286	4	27
98	1	89.36188171	4	25
85	1	115.4431649	4	27
83	1	99.44963944	3	25
115	1	92.74571	4	27
98	0	107.9018378	1	18
69	1	138.8931061	4	20
106	1	125.1451195	3	23

Predicted outcomes:

100
90
81
55
84
98
82
100

“True” outcomes:

102
95
91
58
84
78
102
110

3) Compare predicted with true outcomes to evaluate model performance.

Testing set:
<predictors>, model outputs predictions

1	81.61952618	1	24
1	95.07306862	1	19
1	88.57699772	1	23
1	94.85970819	4	24
1	88.96280085	4	27
1	114.114297	4	26
0	100.5340719	2	24
1	120.4191456	1	26

1) Model is “fit” to the training data.

2) Model is given test data, outputs its predictions.

Linear Regression

Assume a linear model can describe the data.
The general model formula (each "*i*" is a row of data):

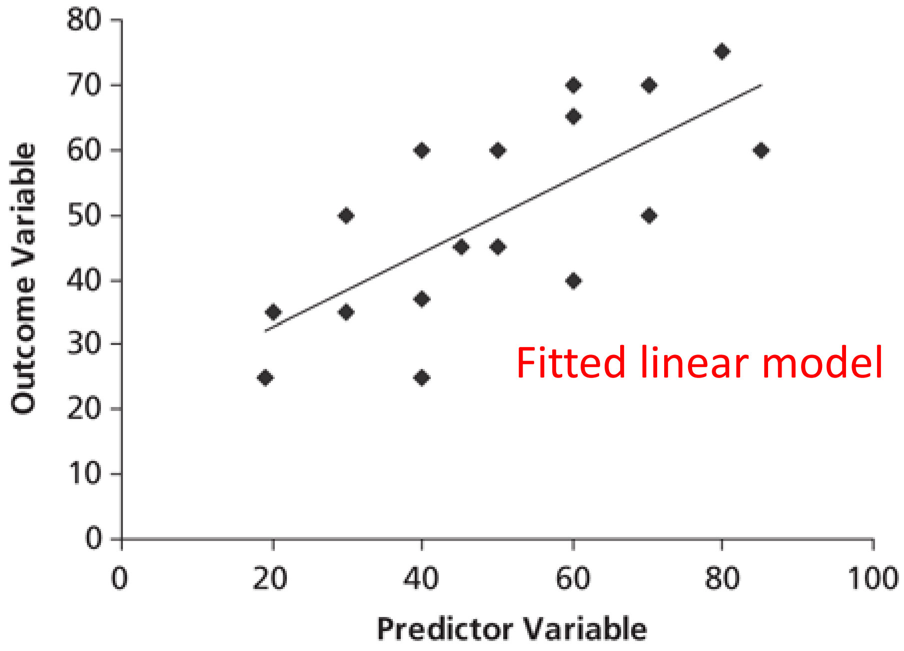
$$y_i = \alpha + \beta x_i + \varepsilon_i$$

Error, or "noise" term

Outcome

Parameters

Predictor



The formula is a linear combination of the input variables (predictors). Notice that the formula Above is the equation of a line. The output of the model is a real-valued number.

Linear Regression

Very commonly used for exploratory and confirmatory analysis.

Major Assumptions:

- The relationship between the covariates and response is linear.
- All covariates have the same variance.
- The covariates do not interact.
- There are others…

Terminology:
Terms used to describe the data used by the model:
input variables, predictors, covariates, independent variable.
Terms used for the predicted variable:
outcome, response, dependent variable.

Multiple Linear Regression

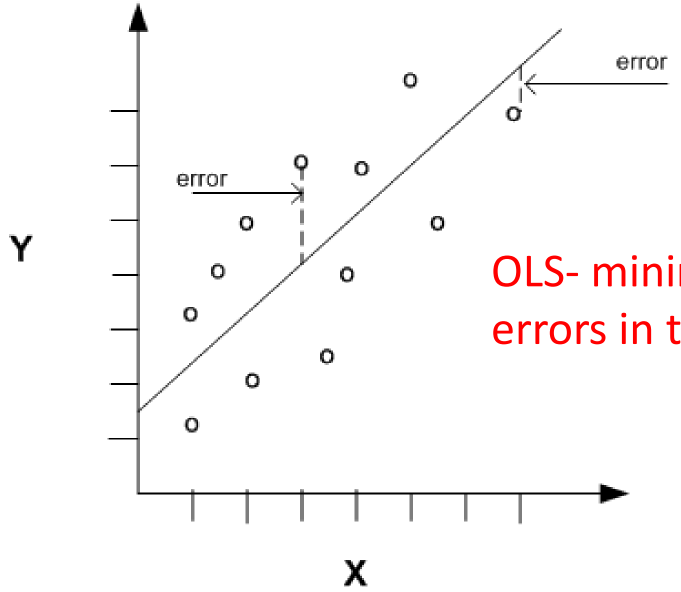
- When there is more than one input variable we have multiple linear regression. The linear formula for *n* input variables:

$$\hat{y}_i = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \beta_3 x_3 + \ldots + \beta_i x_i + \ldots + \beta_n x_n$$

- Notice that there are *n*+1 parameters in the model, one coefficient for each input, and one intercept parameter.
- The parameters are typically represented by greek letters, often beta or theta.
- When the model is fitted to training data, it learns the values of the parameters that minimize the fit errors (residuals).

Linear Regression- fitting

- The typical algorithm used to learn the parameters is Ordinary Least Squares, OLS.
- We'll not go into the details, but think of it as finding a straight line through the data points that minimizes the errors, the distance between each data point and the line.



OLS- minimize the errors in the fit.

- Of course, with multiple linear regression the dimensions of the space are equal to the number of input variables.

Linear Regression- fitting example

Predicting child cognitive test score from data about mom's IQ, High school completion, time spent at work, age.

The model below is presented in an R formula format.

Example R output of fitting a multiple Linear regression model to a data set With 4 input variables.

The outcome to be predicted

The input variables

```
Call:
lm(formula = kid_score ~ mom_hs + mom_iq + mom_work + mom_age +
    mom_hs:mom_iq, data = train.data)
```

The training data set

Note that some variables are numeric, such as mom_iq and age, while some are categorical, such as mom_hs: either she graduated from HS or not: "yes"/"no". This can be represented by 1 or 0.
For factors with multiple levels, several binary variables can be used.
Thus, linear regression can handle categorical data if it is encoded as numbers.

Linear Regression- fitting example

Predicting child cognitive test score from data about mom's IQ,
High school completion, time spent at work, age.

The model below is presented in an R formula format.

Example R output of fitting a multiple
Linear regression model to a data set
With 4 input variables.

The outcome to be predicted

The input variables

```
Call:
lm(formula = kid_score ~ mom_hs + mom_iq + mom_work + mom_age +
    mom_hs:mom_iq, data = train.data)
```

The training data set

An "interaction" term. If you think that high school completion and IQ are correlated
This can help the model be more flexible, but we don't want too much flexibility.
Why? Overfitting!

Linear Regression- evaluate fit

R output as a result of execution a fit of the model to the training data (calling the R function “lm”).

```
lm(formula = kid_score ~ mom_hs + mom_iq + mom_work + mom_age +
    mom_hs:mom_iq, data = train.data)

Coefficients:
              Estimate Std. Error t value Pr(>|t|)
(Intercept)   -33.4940    17.7890  -1.883 0.060580 .
mom_hs         58.0990    16.9437   3.429 0.000681 ***
mom_iq          1.0522     0.1601   6.573 1.87e-10 ***
mom_work       -0.4377     0.8358  -0.524 0.600826
mom_age         0.6974     0.3731   1.869 0.062453 .
mom_hs:mom_iq  -0.5603     0.1760  -3.183 0.001592 **
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 17.76 on 338 degrees of freedom
Multiple R-squared:  0.2529,    Adjusted R-squared:  0.2418
F-statistic: 22.88 on 5 and 338 DF,  p-value: < 2.2e-16
```

The parameters learned from fitting the model

*** means highly significant, i.e. the p-value from a t-test indicates it is highly unlikely that the true value of this coefficient is 0.

Significance symbols defined here.

The coefficients- one for each input variable (plus an intercept).
Positive sign indicates positive correlation with outcome- negative indicates a negative relationship.
Numbers: mom_work for example, for each increase in one unit of mom_work, the output will go down 0.4377 units on average, holding all other inputs constant.

Linear Regression- evaluate fit

```
Call:
lm(formula = kid_score ~ mom_hs + mom_iq + mom_work + mom_age +
    mom_hs:mom_iq, data = train.data)

Coefficients:
               Estimate Std. Error t value Pr(>|t|)
(Intercept)    -33.4940    17.7890  -1.883 0.060580 .
mom_hs          58.0990    16.9437   3.429 0.000681 ***
mom_iq           1.0522     0.1601   6.573 1.87e-10 ***
mom_work        -0.4377     0.8358  -0.524 0.600826
mom_age          0.6974     0.3731   1.869 0.062453 .
mom_hs:mom_iq   -0.5603     0.1760  -3.183 0.001592 **
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 17.76 on 338 degrees of freedom
Multiple R-squared:  0.2529,	Adjusted R-squared:  0.2418
F-statistic: 22.88 on 5 and 338 DF,  p-value: < 2.2e-16
```

The SD of the fit errors- the scale of the residuals.
The average distance a prediction would fall from a true value.

R-squared- the fraction of variance the model "explains". Adjusted is probably more realistic. Here, about 25% of variance explained by model, which is fairly disappointing.

DF=number of data points – number of estimated coefficients.

F-test compares a model with no predictors (an intercept-only model) to the model fitted.
Null hypothesis: The fit of the intercept-only model and your model are equal.
Alternative hypothesis: The fit of the intercept-only model is significantly reduced compared to your model.

Linear Regression- evaluate fit

R output as a result of execution a fit of the model to the training data (calling the R function “lm”).

```
Call:
lm(formula = kid_score ~ mom_hs + mom_iq + mom_work + mom_age +
    mom_hs:mom_iq, data = train.data)

Coefficients:
              Estimate Std. Error t value Pr(>|t|)
(Intercept)   -33.4940    17.7890  -1.883 0.060580 .
mom_hs         58.0990    16.9437   3.429 0.000681 ***
mom_iq          1.0522     0.1601   6.573 1.87e-10 ***
mom_work       -0.4377     0.8358  -0.524 0.600826
mom_age         0.6974     0.3731   1.869 0.062453 .
mom_hs:mom_iq  -0.5603     0.1760  -3.183 0.001592 **
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 17.76 on 338 degrees of freedom
Multiple R-squared:  0.2529,	Adjusted R-squared:  0.2418
F-statistic: 22.88 on 5 and 338 DF,  p-value: < 2.2e-16
```

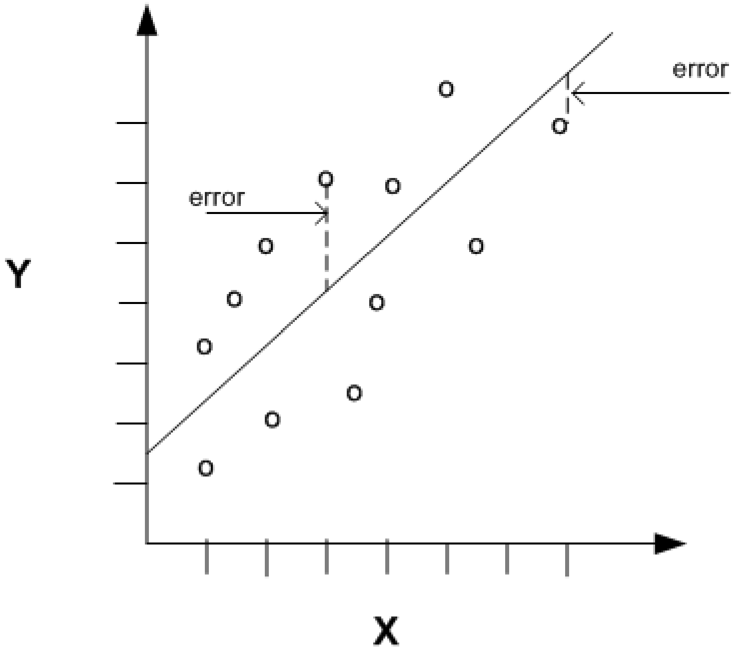
Be careful evaluating a model fit.
The model should fit well, with a relatively low std error and have some significant variables.

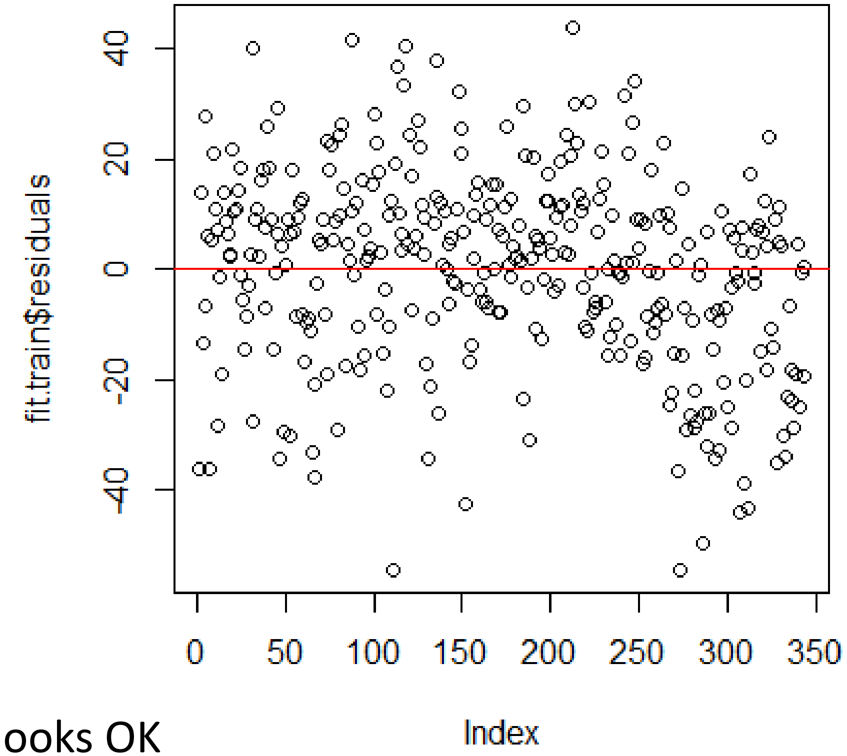
We are usually comparing several models
To each other to find the best model.

Think: there might be a better model, or, the data may not have a clear trend.

Fit errors

Another check is to plot the fit errors, or residuals. If the fit is good, the errors should be Fairly evenly distributed around 0.





This residual plot looks OK

Model selection

- Model selection is the process of specifying the “best” model- one that fits the data reasonably well and does a good job predicting an outcome on new data.
- In the previous example, this model was used:

```
Call:
lm(formula = kid_score ~ mom_hs + mom_iq + mom_work + mom_age +
    mom_hs:mom_iq, data = train.data)
```

- Perhaps there is a simpler model that would work just as well if not better?
- Note: removing or adding variables to the model changes the coefficients for all variables.

Model selection

- You can manually try models with different combinations of variables, evaluating their fit and predictive performance.

- You can also use an automated method for model selection. These algorithms typically use a “penalty” term to avoid overfitting with large, complex models.

- An example of such a penalty term is the Akaike information Criterion, or AIC.

$$AIC = 2k - 2\ln(L)$$

- Where L is the maximum value of the likelihood function for the model, and k is the number of estimated parameters in the model.

- The term for penalizing a complex model to avoid overfitting is called “regularization”.

Model Evaluation

- Once a model has been selected and fitted, it can be evaluated for predictive performance.
- In linear regression, we are dealing with numbers as output from the model.

One common measure is MSE, or Mean squared error:

"True" outcomes:	Predicted outcomes:
102	100
95	90
91	81
58	55
84	84
78	98
102	82
110	100

$$MSE = avg\left((true\ outcomes - predicted\ outcomes)^2\right)$$

Linear regression

- Used widely- easy to generate, fairly simple, easy to interpret.
- Helps understand how variables interact with outcome and each other.
- Can use categorical (with modifications) and numeric inputs.
- Predicts real-valued outcomes.
- Assume predictors have a linear relationship with outcome.
- Best used in combination with other methods to get the results of different “points of view”.