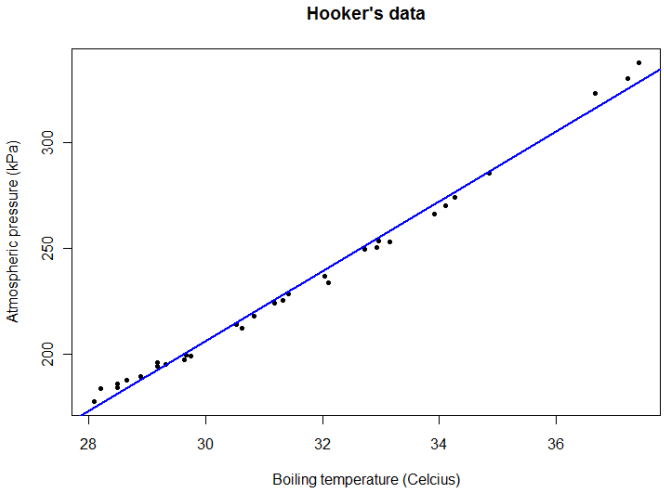
HEC MONTRÉAL

Theme 3

Principles of supervised learning
(with linear regression)

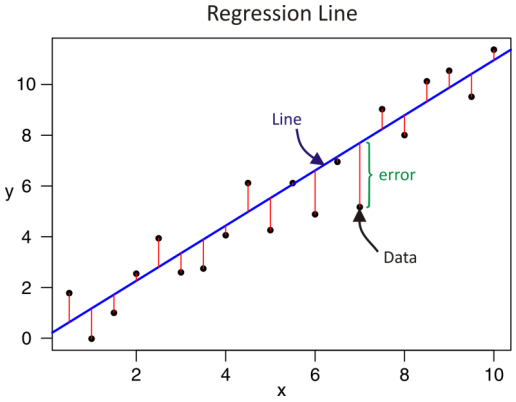
Historical example



- Collected by Hooker and published in a 1957 article.
- Model predicts atmospheric pressure from temperature
- Thermometer cheaper and less fragile than a barometer.

Linear regression: Which line is the best line?

The best line is defined as that minimizing the sum of squared errors (least squares)



Multiple linear regression:

- Best hyperplane between many variables X and one continuous$Y$.
- Model independent observations: $Y = \beta_0 + \beta_1 X_1 + \beta_2 X_2 + \cdots + \beta_p X_p + \varepsilon$.
- Where $\varepsilon \sim N(0, \sigma^2)$

Mathematical derivations

Finding the parameters of the regression mathematically is not hard. You need to write down the model, and optimize it (derive, solve those equations = 0).

With a sample of n values $y_1, \dots, y_n$ and only one covariate $x_1, \dots, x_n$ for instance, we want to minimize

$$\sum_{i=1}^{n} \{y_i - (\beta_0 + \beta_1 x_i)\}^2$$

A page of calculus will give you the formulas for the parameters.

Matrix notation

For multiple regression, it is easier and more compact to adopt a matrix notation.

$\boldsymbol{Y}$ (the target variable) is a vector of n elements.

The design matrix $\boldsymbol{X} = \left[\boldsymbol{1}|\boldsymbol{X_1}|\ldots|\boldsymbol{X_p}\right]$ is formed by placing the covariates one per column. In that context, the intercept is in fact a column of ones.

The model can then be written as $\boldsymbol{Y} = \boldsymbol{X\beta} + \boldsymbol{\varepsilon}$ where $\boldsymbol{\beta}$ is a vector of length $p + 1$ and $\boldsymbol{\varepsilon}$ is a multivariate normal vector with a diagonal covariance matrix $\sigma^2\boldsymbol{I}$.

Then, simple calculus yields the estimates we need:

$$\widehat{\boldsymbol{\beta}} = (\boldsymbol{X}'\boldsymbol{X})^{-1}\boldsymbol{X}'\boldsymbol{Y}$$

The predictions for Y is a projection.

$$\widehat{\boldsymbol{Y}} = \boldsymbol{X}\widehat{\boldsymbol{\beta}} = \boldsymbol{X}(\boldsymbol{X}'\boldsymbol{X})^{-1}\boldsymbol{X}'\boldsymbol{Y} = \boldsymbol{H}\boldsymbol{Y}$$

Where the hat matrix $\boldsymbol{H}$ is a projector of $\boldsymbol{Y}$ unto the span of the design matrix.

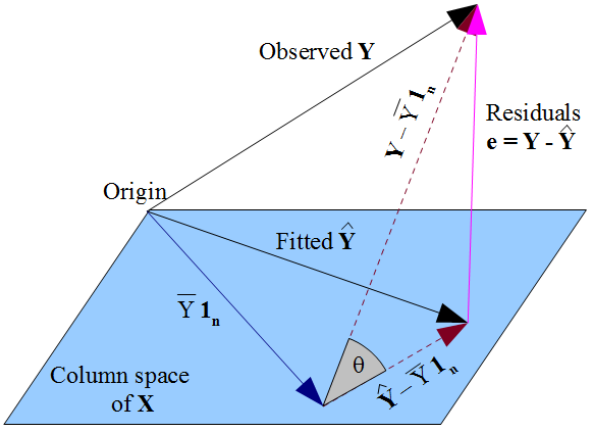
Geometry of regression

In an n dimensional space, the vector $\boldsymbol{Y}$ is projected unto the space spanned by the columns of $\boldsymbol{X}$, a $p+1$ dimensional sub-space.

Understanding the geometry of regression can help better grasp issues such as Issues such as multicollinearity and variable selection.

- The contribution of a new variable depends on the dimension the previous design space
- Multicollinearity happens when some variables in the design space are too correlated. These variables are like coordinates in that space.

For inference this geometric view also explains why some sums of squares are independent.



The art of making predictions

Statistical models may be used for:

- Inference: understanding and proving the nature of things,
- Prediction: generating the best possible predictions, even if they are not based on a causal link.

Supervised learning: focus is on prediction, not inference.

Possible predictions:

- How much is this potential client worth?
- Which clients are most likely to cease business with us?
- What potential donors should I contact from my list of members?
- What is the value of an item?
- How many warranty claims should I expect next month?

To make predictions on, say $Y =$ "sales next week", you need

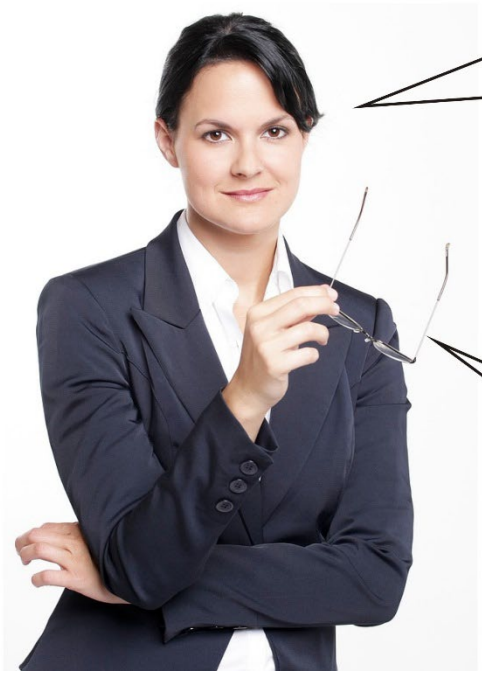
- Historical **data** on Y as well as other variables X that can be predictors.
- The predictors:
  - Are known in the historical data,
  - Will be known for future events when we need them.

Challenge: making predictions for Y that are unknown or not realized yet.

One solution is to:

- Fit a linear regression on historical data,
- Use the equation of the regression to make predictions.

2009 2010 2011 2012 2013 2014 2015 **2016** 2017 2018

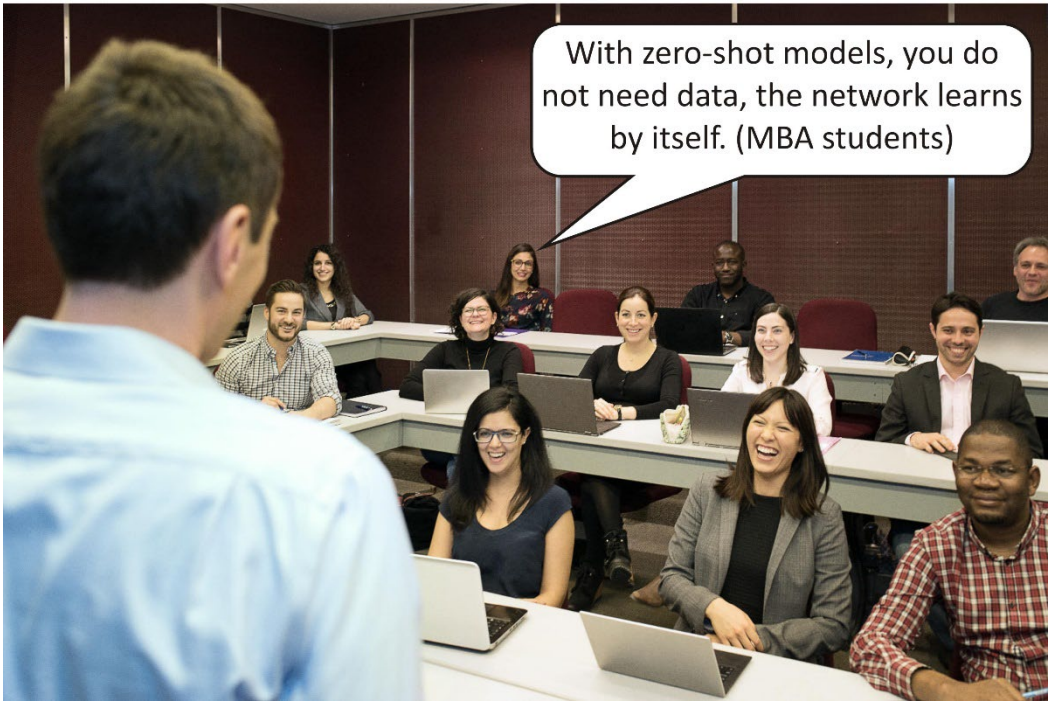


I have data on individuals and want to make suggestions for them on restaurants or shows when they visit a new city.

What data do you have?

Library fines, parking tickets, but only some.

2009 2010 2011 2012 2013 2014 2015 2016 2017 **2018**



Scenarios

Which of the suggested variables could be used as predictors in each scenario?

Predict the box-office of a movie one week before its release.

$Y =$ box-office of different movies for their first weekend.

- Number of screens,
- Advertising budget,
- Type of movie,
- Release date,
- Simultaneous release of a blockbuster,
- Intensity of the social media "buzz" on the release date,
- Visibility in traditional media during the week leading to the release,
- Amount of rain during the first weekend,
- Presence of some stars in the casting.

Predictive maintenance

$Y =$ pressure on an air filter in and HVAC, recorded by sensors every 15 minutes over a period of 1.5 years.

Must decide when to clean or replace the filter while avoiding unnecessary maintenance. What will be the pressure next week if we do not change the filter?

Possible variables:

- Outside temperature,
- Number of days since the last cleaning,
- Number of days since the last replacement,
- Pressure from other sensors in the system,
- Date,
- Holiday or not.

Example

The price of a diamond depends on its characteristics, the four Cs:

- Carat (weight)
- Clarity (from I3 to F)
- Color (Z to D)
- Cut (poor to excellent)

We get the value of 500 diamonds of clarity “VS2” and color “I”.
You are in the market for such a diamond (for your significant one).

Variable	Description
price	Price in US dollars
carat	Weight of the diamond
cut	Quality of the cut (Fair, Good, Very Good, Premium, Ideal)
x	Length in mm
y	Width in mm
z	Depth in mm

Before starting to use a dataset, it is very important to explore it, making sure that it contains the right data and that it was read correctly.

Discussions about that are found in the accompanying material.

```
> str(train)
Classes ‘tbl_df’, ‘tbl’ and 'data.frame':       500 obs. of  11 variables:
 $ carat  : num  1.6 0.69 1.71 1.06 1.22 1.2 0.78 0.71 1.58 1.28 ...
 $ cut    : Ord.factor w/ 5 levels "Fair"<"Good"<..: 3 3 5 5 4 5 5 5 4 5 ...
 $ color  : Ord.factor w/ 7 levels "D"<"E"<"F"<"G"<..: 6 6 6 6 6 6 6 6 6 6 ...
 $ clarity: Ord.factor w/ 8 levels "I1"<"SI2"<"SI1"<..: 4 4 4 4 4 4 4 4 4 4 ...
 $ depth  : num  60 63.3 60.5 61.4 59.3 61.5 61.2 61.5 59.4 61.6 ...
 $ table  : num  60 57 56 57 59 57 58 55 61 57 ...
 $ price  : int  10497 2070 11559 4405 6156 5880 2652 2878 8583 6838 ...
 $ x      : num  7.55 5.64 7.73 6.57 7.01 6.8 5.88 5.76 7.66 7.01 ...
 $ y      : num  7.59 5.57 7.71 6.52 6.96 6.75 5.92 5.78 7.63 6.98 ...
 $ z      : num  4.54 3.55 4.67 4.02 4.14 4.17 3.61 3.55 4.54 4.31 ...
 $ carat2 : num  2.56 0.476 2.924 1.124 1.488 ...
```

You are considering a diamond of 1.33 carats that costs $6,203.

Is this a good deal?

Let us first try to predict its value based on weights alone.

Parameter	Estimate	Standard Error	t Value	Pr > |t|
Intercept	-3694.063261	125.9458541	-29.33	<.0001
carat	8765.742756	106.5065242	82.30	<.0001

Yielding the equation

$$\text{Price} = -3694.06 + 8765.74\ \text{carat}$$

This model predicts **$7,964.37**. What a deal!

What happens if we also take into consideration the quality of the cut?

The cut of that 1.33 carat diamond is “very good”.

Parameter	Estimate		Standard Error	t Value	Pr > |t|
Intercept	-4580.520102	B	301.4215840	-15.20	<.0001
carat	8764.894175		107.3012610	81.68	<.0001
cut Good	588.797659	B	332.4754120	1.77	0.0772
cut Ideal	970.760801	B	300.1977600	3.23	0.0013
cut Premium	875.526631	B	307.7095528	2.85	0.0046
cut Very Good	1048.959815	B	307.3933091	3.41	0.0007
cut Fair	0.000000	B	.	.	.

$$\text{Price} = -4580.52 + 8764.89\ \text{carat} + 588.80\ \text{Good} + 1048.96\ \text{VeryGood} + 875.53\ \text{Premium} + 970.76\ \text{Ideal}$$

Each of the cut variables is equal to one only when the diamond is of that cut.

The predicted value is **$8,125.75**. Wow! What a deal at $6,203!

Les us consider the other measurements of the diamond, namely $(x, y, z) = (6.99, 7.02, 4.38)$.

Parameter	Estimate		Standard Error	t Value	Pr > |t|
Intercept	14337.86006	B	714.2163166	20.07	<.0001
carat	18511.00679		363.5596033	50.92	<.0001
cut Good	-104.31338	B	212.9245915	-0.49	0.6244
cut Ideal	334.18775	B	193.2358855	1.73	0.0844
cut Premium	80.38934	B	194.9046077	0.41	0.6802
cut Very Good	253.83564	B	201.4896150	1.26	0.2083
cut Fair	0.00000	B	.	.	.
x	-3994.43580		712.9919402	-5.60	<.0001
y	1885.70240		684.0912331	2.76	0.0061
z	-3892.15904		361.1908038	-10.78	<.0001

The predicted value is then **$7,480.20**.

So what is the value of that diamond???

Is it **$7,480.20**, **$7,964.37** or **$8,125.75**, or the asking price?

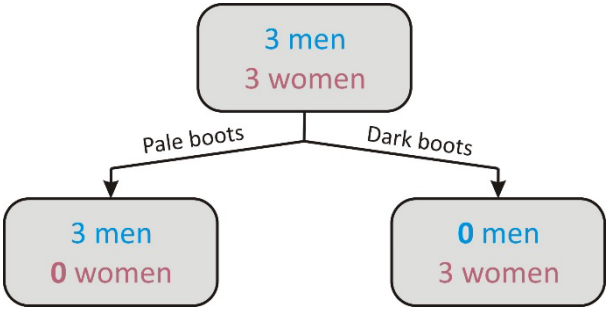
All the models considered offer a reasonable predictions.
None suggests $500 or $15,000.

Predictive models all make errors. The challenge is to evaluate the magnitude of those error and to find a model that yields the smallest possible errors.

Evaluating a model

We need to tell men from women (sample of HEC Montréal's students below). Let's build a model and consider its misclassification rate.

I suggest a classification tree:

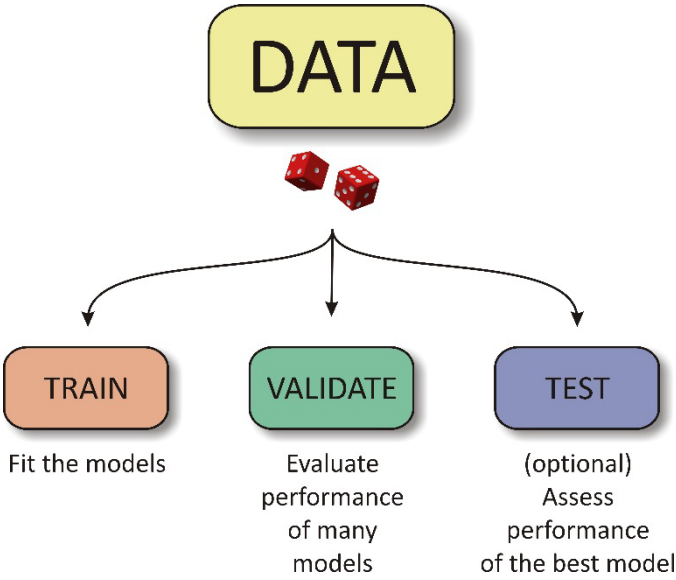


Misclassification rate: 0%

Can we get any better than that?

When we fit a model, we choose the parameters that provide the best fit… for the training data. If we use the same data to assess performance, the model will look better than it is in reality.

Training – Validation – Test



NEVER evaluate performance on data that was used to train.

A sample of 200 diamonds was removed initially and may now be used for validation.

Measure of performance

When predicting a continuous value, we would like our model to be as close as possible to the truth, or in other words, have a low *Generalized Mean Square Error*:

$$GMSE = E[\{Y - \widehat{f_Y}(X)\}^2]$$

Where Y is the variable we want to predict, $\widehat{f_Y}(X)$ is the prediction from the model with covariates X, and the expectation means that we want an average over the (imaginary) infinite population. In practice, we estimate the *GMSE* with the mean squared error, or more often we use its square root, the *Root Mean Squared Error*:

$$RMSE = \sqrt{\frac{1}{\text{size of validation set}} \sum_{\text{validation data}} (\text{real price} - \text{predicted price})^2}$$

Which model should we choose?

Let us consider the three models:

Model	RMSE
Carat only	1398.858022
Carat and cut	1404.602926
All variables	822.152608

Note that:

- Adding the variable “cut” decreased the performance.
- The best model here (by a large margin) contains all variables.

The RMSE found for the model chosen as “best among many means” will typically be higher than its true value: if many models are similar, the luckiest (for having an especially small RMSE) among the pretty good ones will be picked.
Estimating the RMSE on the test set will provide a good estimate of RMSE.

Should we always use all the variables?

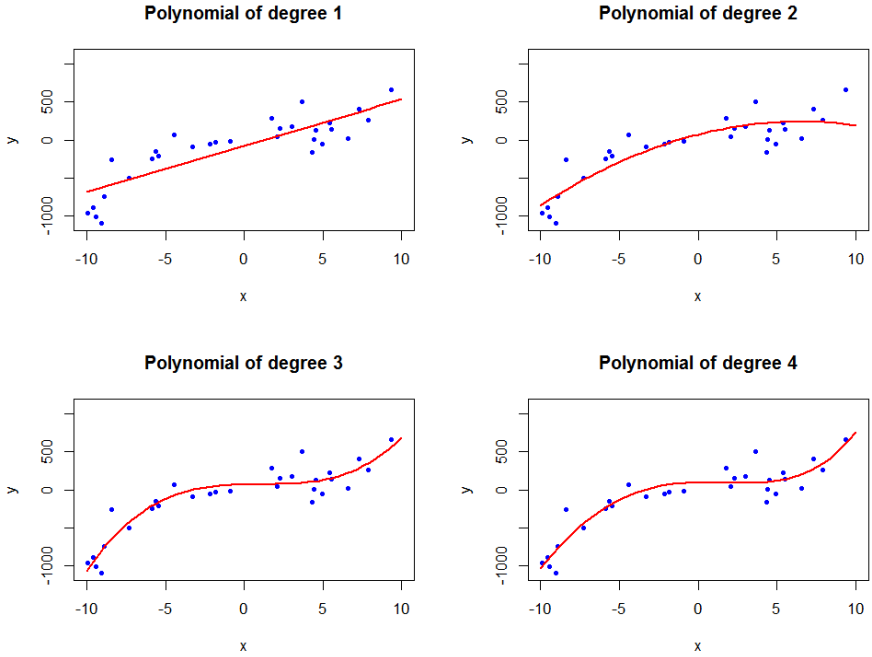
NO! Adding variables may improve performances, but adding useless variables degrades the performance by adding noise to the model through **overfitting**.

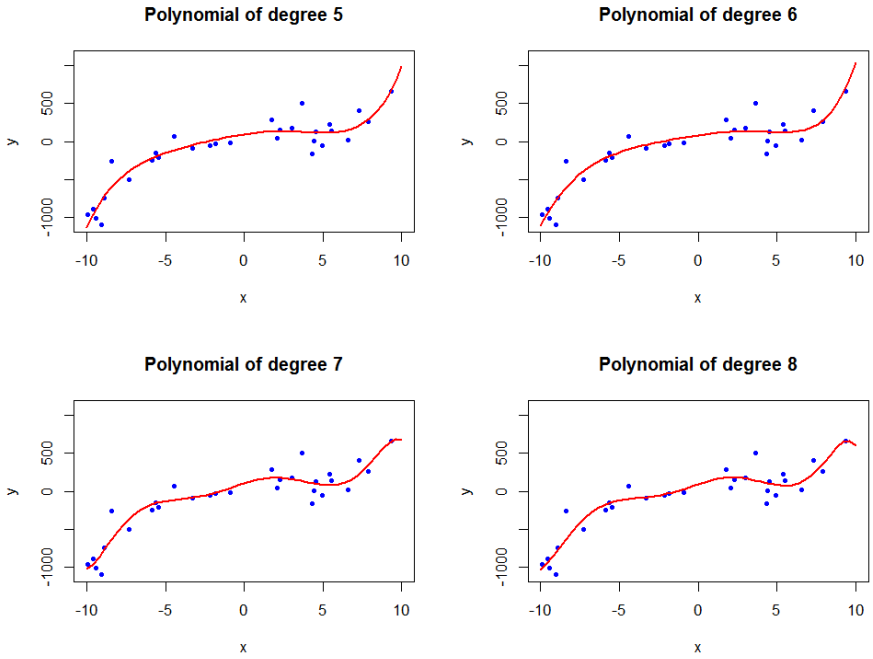
Consider a special case of linear regression where powers of a single explanatory variable (X) are added to form a polynomial function.

We simulate the following "true" model (known because we generate the data):

$$Y = X^3 - X^2 - 6X + \varepsilon$$

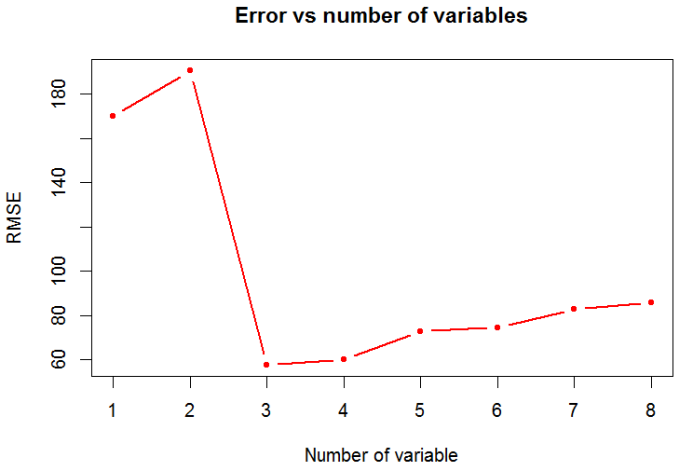
We regress Y on$X$, X^2, X^3, X^4, etc. and interpret the results as a polynomial:





As the number of variables increase, the model adapts to peculiar characteristics of the data (noise) rather than capturing the general shape of the model.

In fact, here is the RMSE when comparing to the true model:

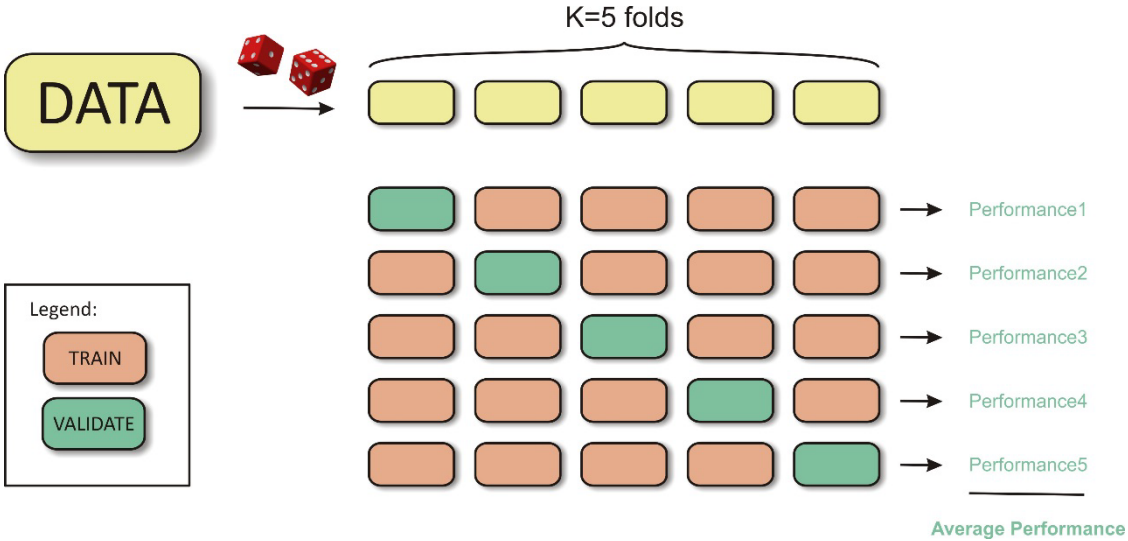


Nb of variables	RMSE
1	169.90
2	190.46
3	**57.79**
4	60.19
5	72.92
6	74.63
7	82.90
8	85.86

- Error decreases quickly, but starts increasing thereafter.
- Adding useless variables decreases performances.
- Strategy :
  - Consider as many variables as possible,
  - Chose the best subset among them.

Cross-validation

When there are not enough data to leave a validation set, cross-validation allows to never evaluate a model on the training data, yet use all data to fit in the end.



- Each model hence gets a measure of performance.
- The chosen model is fitted on the whole dataset to get its final parameters.

Penalized measures of fit

A measure the performance calculate on the training data will always improve when we add variables in the model. This is how some statistics were developed to account for the number of parameters (k) in a model fitted on n data.

- $R^2_{aj} = 1 - \frac{n-1}{n-p-1}(1 - R^2)$

Is an adjustment to the R^2 statistic. A bigger R^2_{aj} is better.

The maximum likelihood estimate for the linear regression model is the least squares! The next statistics may in fact be used for any model fitted with maximum likelihood.

- $AIC = 2\,k - 2\ln(lik)$
- $BIC = k\ln(n) - 2\ln(lik)$

where lik is the value of the likelihood at its maximum.
Smaller is better for AIC and BIC.

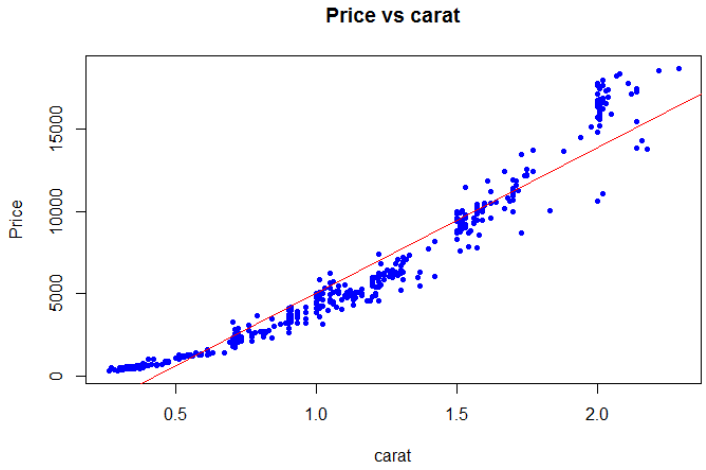
These three values may identify different ideal models.
None corrects as well as using a hold-out validation sample.

Creating variables

We can create additional variables from the ones we have by:

- Adding their power,
- Adding interaction terms (multiplication of two predictors)

In the diamond example, the link between carat and price does not seem to be linear.



Model	RMSE
Carat only	1398.858022
Carat and cut	1404.602926
All except carat^2	822.152608
Carat and carat^2	810.846954
Carat, carat^2 and cut	800.785994
All with carat ^{2}	778.446878

Note that:

- The model with carat and carat^2 performs better than the model with all variables except carat^2.
- Adding carat^2 improves the performance of the models considered previously.
- The "All with carat^2" model predicts a value of $7,208.49 for the 1.33 carats diamond considered.

Model selection

With p predictors, one can decide whether or not each predictor is used.
Considering all possible models is a good idea, but it quickly gets out of control:

Number of variables (p)	Number of models	Number of models (with interactions)
2	4	5
5	32	1450
10	1 024	35.883×10^{12}
20	1 048 576	1.569×10^{57}

When possible, all possible subsets can be considered. The package *leaps* can do that for moderate numbers of variables.

Otherwise, model selection methods have been developed to explore the set of all possible models without computing them all.

Feature selection

Classic solution: construct models by adding and/or removing variables one at a time.

Backward selection:

1. Start with the complete model (all variables),
2. Fit all models with one less variable,
3. Identify the lowest AIC,
4. If the new AIC is larger than before, stop.
   Otherwise repeat from step 2 with the new model.

A classical approach uses p-values to identify variables to dismiss and control the complexity of the model. Nowadays, criteria such as AIC and BIC are preferred.

Forward selection:
Same idea, but starting empty and adding variables one by one.

S*tepwise*:
Alternating backward and forward steps.

Example 2: Wine quality

Let us consider another dataset on white wine. We have 11 characteristics (fixed acidity, volatile acidity, citric acid, residual sugar, chlorides, free sulfur dioxide, total sulfur dioxide, density, pH, sulphates, alcohol) as well as a **score of quality**.

Training set: a list of 1,500 white wines.
Validation set: Another list of 1,500 wines.
Test set: A third set of 1,000 wines.

Let us consider different models, including:

- Some that are manually picked
- Backward selection from the original variables,
- Stepwise selection from all variables and their interactions.

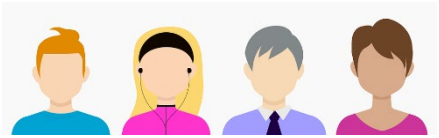
Model	p	RMSE (valid)	RMSE (test)
Alcohol	1	0.8034	0.7937
Fixed, volative, sugar, alcohol	4	0.7609	0.7536
Fixed, volative, sugar, density, pH, sulphate	6	0.7601	0.7500
All variables	11	0.7534	0.7444
Stepwise from all variables	7	0.7533	0.7430
All subsets using BIC	5	0.7573	0.7478
All subset validation RMSE	8	0.7493	0.7384
All variables and interactions	77	0.7407	**0.7309**
Stepwise from all variables and interactions	41	**0.7349**	0.7321

- Stepwise model: better than all variables (barely) but not as good as adding interactions.
- Stepwise on all variables + interactions win on validation, but not on test.
- Too long to compute all subsets on validation RMSE…

How is this useful?

In this example, we choose better wine, but we could also identify:

- more valuable clients,
- better products,
- most efficient contractors,
- etc.



Consider this scenario (with the white wine test set):

A distributor offers 1,000 assembly wines and provides the chemical variables. Since the wines are assembled, it is impossible to rely on grape variety or regions. We are able to test only **50** wines. Which one should we choose?

Option 1: Random choice

- In the test set, 326 of the 1,000 wines have a quality of 5 or less (*piquette*).
- We expect approximately 16 bottles of *piquette* out of 50.
- Only 216 wines are of top quality (7 or more).
- We expect 11 bottles of top wine.
- The average quality of the 1,000 wines is 5.88, we therefore expect a similar average quality for the 50 wines.

Option 2: Use a model

- The model predicts the quality of the 1,000 wines (normally unknown).
- We can choose the 50 wines with the highest predicted quality.
- Will we have fewer than 16 bottles of *piquette*?
- More than 11 bottles of top wine?
- Will average quality exceed 5.88?

Model	Average Quality	# Piquette	# Top wine
Random	5.88	16	11
Alcohol	6.68	1	30
Fixed, volative, sugar, alcohol	6.58	0	24
Fixed, volative, sugar, density, pH, sulphate	6.64	1	27
All variables	6.70	1	29
Stepwise from all variables	6.70	1	29
All subsets using BIC	6.62	1	25
All subset validation RMSE	6.76	1	30
All variables and interactions	6.76	2	34
Stepwise from all variables and interactions	6.72	3	33

- With a model, you get much better wine without tasting it all.
- Similarly, a company can identify more profitable or more loyal clients.