(MSc) Data Mining

Topic 04 : Regression

Part 01 : Overview

Dr Bernard Butler and Dr Kieran Murphy

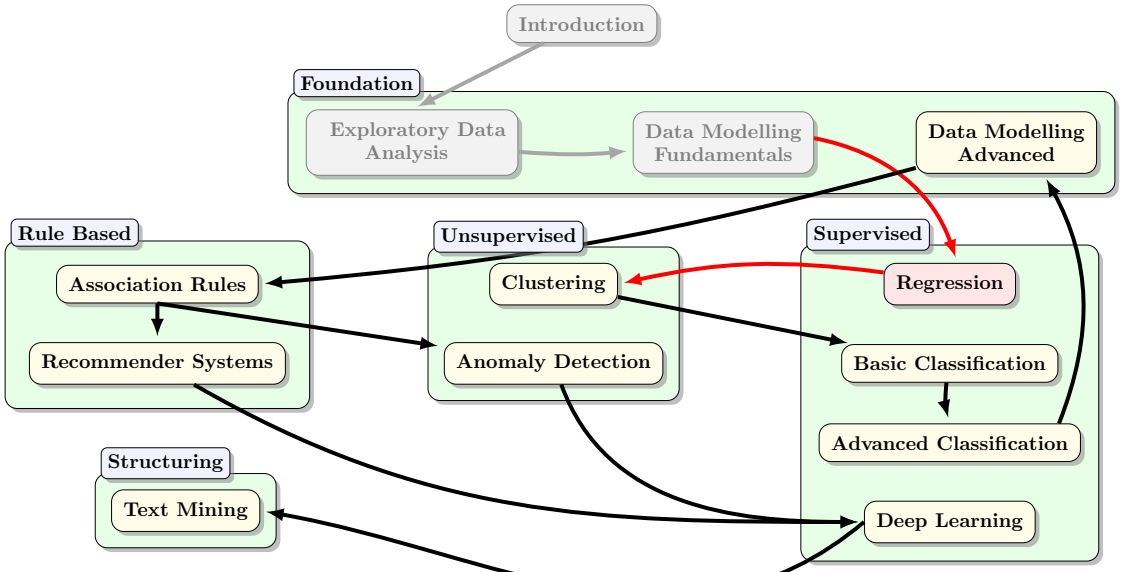
Department of Computing and Mathematics, SETU Waterford.
(bernard.butler@setu.ie; kmurphy@wit.ie)

Spring Semester, 2025

Outline

- Regression as a means of minimising sum of the squared errors
- Regression assumptions - what they mean, how they can be used for validation and model building
- Case studies from Advertising, Diamond sales, Credit Balance prediction

Data Mining (Week 4)



Overview — Summary

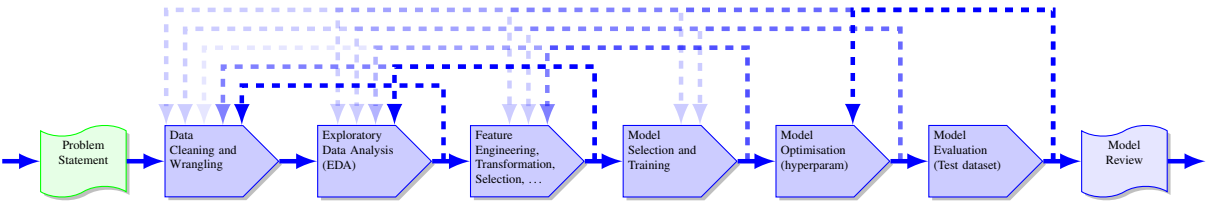
This Week's Aim

This week's aim is to give an overview of linear regression: fitting linear models to data, to predict a numeric value.

- High level view of regression: where it came from, what it attempts to do.
- Examine some extensions to the simplest case of linear regression.
- Consider how to check that the regression was successful, and make some improvements if necessary
- To provide context we will use the following datasets over this and Regression 2:
  - Generated data (various)
  - Diamond dataset: predicting diamond prices given their weights
  - Advertising dataset: predicting widgets sold based on spending in different advertising channels
  - Credit dataset: predicting credit balance using income, status, etc.

If you need to predict a categorical value (such as whether a passenger survived the sinking of the Titanic), that is classification, not regression.

Data Mining Workflow: where Regression Modelling fits



- Data pre-precessing (Data Cleaning + EDA + (Basic) Feature Engineering) is the foundation of Data Mining.
- Train-test split, feature engineering and reducing errors (covered by Kieran last week) play key roles in Regression.
- Data Models are needed for prediction: learning from training data, so that predictions can be made on new data.
- Regression models have common characteristics and assumptions - we consider them today.
- Regression is the process of building, improving, validating and visualising such models.

But first - where did the term come from?

"Regression to the Mean"

- Sir Francis Galton (1822-1911): polymath, genius, eugenicist
- Found that children of tall parents "regressed" towards average height of population
  - Let δh be the height difference between their parents and the general population.
  - Then the height difference of the children versus their peers is predicted to be $\frac{2}{3}\delta h$.
  - So: the height of children can be predicted using a simple formula (mathematical expression).
- Techniques used, and underlying statistics, were applicable to other problems, so the name stuck.
- Regression has been expanded to handle larger and more complex data.
- Galton "fitted" the formula by eye, but Legendre and Gauss previously (early 1800s) invented *least squares* for predicting orbits in astronomy.
- Used to require hours of tedious calculation, but not any more!

Simple Linear Regression: Background

- Linear regression was discovered by Gauss and others around 1800. The "name" came later!
- With small data sets, calculations can be done by hand, but they are tedious and error-prone.
- The goal is simple: Given a training set of (x, y) data where y is assumed to have a linear relationship with x
  - Find the line that is the "best fit" to that data
  - Use the specification of that line to *predict* y for the test x values
- Note that the "linear relationship" of y upon x is just one of the underlying assumptions
- In practice, the data does not have an exact linear relationship, but it should be "close enough"—but what does that mean?
- In terms of Week 3's ML models taxonomy: regression is geometric and parametric
- In terms of Week 3's Components of a Machine Learning Problem
  - Representation is based on (fitting) hyperplanes to point clouds
  - Evaluation usually based on MSE, with assumption checks to help identify the best model family
  - Optimization is one-step (no search needed) because we have a constraint on the errors we allow
- Hyperparameter tuning: polynomial degree, regularisation λ, weights, loss function, ...

Review: Linear combinations and scalar products

Definition 1 (Scalar (dot) product of two vectors)

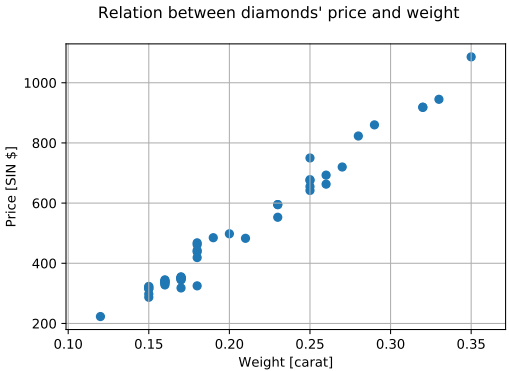
Given two vectors $\mathbf{a}$ and $\mathbf{b}$, each with n elements, the *scalar product* (c) of $\mathbf{a}$ and $\mathbf{b}$ is

$$c \equiv a_1 b_1 + a_2 b_2 + \ldots + a_n b_n = \sum_{i=1}^{n} a_i b_i \equiv |\mathbf{a}||\mathbf{b}| \cos(\mathbf{a}, \mathbf{b})$$

Remarks

- The scalar product of 2 vectors is a scalar, which can be seen as "mixing" two vectors.
- Matrix-vector multiplication $X\mathbf{a}$ can be seen as the scalar product of each row in the matrix X, which is $X(i,:)$ for row i, with the column vector $\mathbf{a}$.
- Alternatively, matrix-vector multiplication can be seen as the *linear combination* of the matrix columns, such as $X(:,j)$, with the column multipliers being the elements of $\mathbf{a}$.
- For linear regression, the matrix columns are the feature vectors $X(j)$ and the column multipliers are the regression parameters $\mathbf{a}$.
- Two nonzero vectors $\mathbf{a}$ and $\mathbf{b}$ can have a scalar product that is zero if $\cos(\mathbf{a}, \mathbf{b}) = 0$, i.e., the $\mathbf{a}$ and $\mathbf{b}$ vectors are perpendicular to each other.

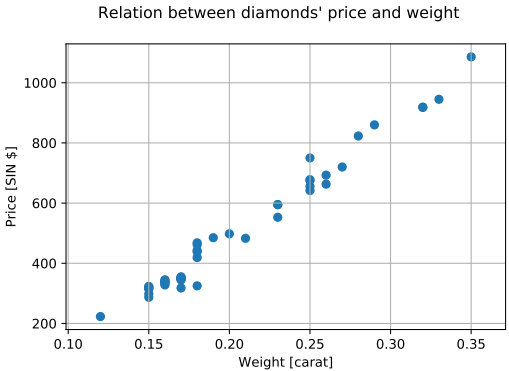
Motivating example: Diamond data



Diamond Prices by Weight

- Given the data on the left, can we use it to predict the price of a diamond that weighs 0.22 carat?
- NB - we have not seen a diamond with that weight before in the data
- Can you think of at least 3 other factors that might affect the price?

Motivating example: Diamond data

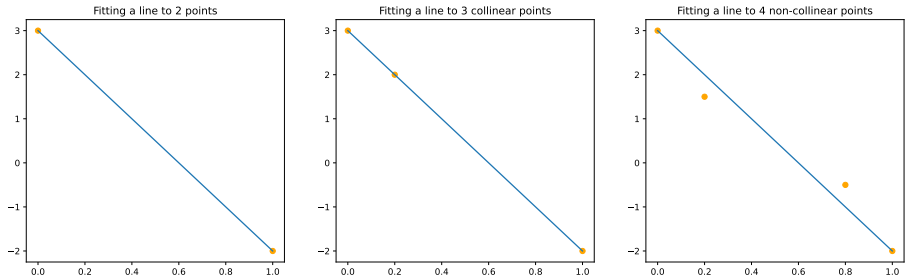


Diamond Prices by Weight

- Given the data on the left, can we use it to predict the price of a diamond that weighs 0.22 carat?
- NB - we have not seen a diamond with that weight before in the data
- Can you think of at least 3 other factors that might affect the price?
- Various(!) - some examples: clarity, cut, provenance, part of a set, ...

Simple Linear Regression: Geometric Intuition

- Given data $\{x_i, y_i\}$ where $i = 2, 3, \ldots, n$ and β_0, β_1 as the (unknown, but to be determined) *intercept* and *slope* of the regression line for this data.
- If $n = 1$, the problem is underdetermined: any line through the point will do - the solution is not unique.
- For $n = 2$ points with $x_2 \neq x_1$, this can be solved uniquely for β_0, β_1, using techniques you learnt for your Junior/Inter Cert.
- For $n > 2$ collinear points, just pick any two points and solve as before.
- Otherwise the problem is overdetermined so need a more general formulation to solve for β_0, β_1.



Simple Linear Regression: Formulation

Definition 2 (Matrix formulation)

- General equation is $y_i = \beta_0 + \beta_1 x_i + \epsilon_i = \hat{y}_i + \epsilon_i$ (data = model + error), where $\hat{y}$ is the predicted y for these values of β_0, β_1.
- Matrix form is $\mathbf{y} = X\beta$. Remember matrix-vector multiplication: inner product of i^{th} row of X times the vector $\beta = 1 \times \beta_0 + x_i \times \beta_1 = \hat{y}_i$.
- However, we don't know β yet, nor do we know $\hat{y}_i$, so we use y_i as an estimate of $\hat{y}_i$ and solve for all data in the training set.
- So: our task is to solve the *overdetermined* (number of rows exceeds the number of columns) system of equations $\mathbf{y} = X\beta$ for β
- Our geometric intuition is that the errors should be "balanced": no benefit to changing intercept (sliding up or down) or slope (tilting the line).

Simple Linear Regression: Normal Equations

$$
\begin{aligned}
\mathbf{y} &\approx X\beta \\
\mathbf{y} &= X\beta + \epsilon \\
X^{\mathsf{T}}\mathbf{y} &= X^{\mathsf{T}}X\beta + X^{\mathsf{T}}\epsilon \\
X^{\mathsf{T}}\mathbf{y} &= X^{\mathsf{T}}X\beta
\end{aligned}
$$

Simple Linear Regression: Normal Equations

$$
\begin{aligned}
\mathbf{y} &\approx X\beta \\
\mathbf{y} &= X\beta + \epsilon \\
X^{\mathsf{T}}\mathbf{y} &= X^{\mathsf{T}}X\beta + X^{\mathsf{T}}\epsilon \\
X^{\mathsf{T}}\mathbf{y} &= X^{\mathsf{T}}X\beta
\end{aligned}
$$

because $X^{\mathsf{T}}\epsilon \equiv 0$ implies the fitted line gives balanced errors and so is 'best". Swapping sides, we have

$$
\begin{aligned}
(X^{\mathsf{T}}X)\beta &= X^{\mathsf{T}}\mathbf{y} \\
(X^{\mathsf{T}}X)^{-1}(X^{\mathsf{T}}X)\beta &= (X^{\mathsf{T}}X)^{-1}X^{\mathsf{T}}\mathbf{y}
\end{aligned}
$$

Simple Linear Regression: Normal Equations

$$\begin{aligned} \mathbf{y} &\approx X\beta \\ \mathbf{y} &= X\beta + \epsilon \\ X^{\mathsf{T}}\mathbf{y} &= X^{\mathsf{T}}X\beta + X^{\mathsf{T}}\epsilon \\ X^{\mathsf{T}}\mathbf{y} &= X^{\mathsf{T}}X\beta \end{aligned}$$

because $X^{\mathsf{T}}\epsilon \equiv 0$ implies the fitted line gives balanced errors and so is 'best". Swapping sides, we have

$$\begin{aligned} (X^{\mathsf{T}}X)\beta &= X^{\mathsf{T}}\mathbf{y} \\ (X^{\mathsf{T}}X)^{-1}(X^{\mathsf{T}}X)\beta &= (X^{\mathsf{T}}X)^{-1}X^{\mathsf{T}}\mathbf{y} \end{aligned}$$

which is equivalent to the *Normal equations*

$$\beta = (X^{\mathsf{T}}X)^{-1}X^{\mathsf{T}}\mathbf{y} \tag{1}$$

Simple Linear Regression: Normal Equations

$$
\begin{aligned}
\mathbf{y} &\approx X\beta \\
\mathbf{y} &= X\beta + \epsilon \\
X^{\mathsf{T}}\mathbf{y} &= X^{\mathsf{T}}X\beta + X^{\mathsf{T}}\epsilon \\
X^{\mathsf{T}}\mathbf{y} &= X^{\mathsf{T}}X\beta
\end{aligned}
$$

because $X^{\mathsf{T}}\epsilon \equiv 0$ implies the fitted line gives balanced errors and so is 'best". Swapping sides, we have

$$
\begin{aligned}
(X^{\mathsf{T}}X)\beta &= X^{\mathsf{T}}\mathbf{y} \\
(X^{\mathsf{T}}X)^{-1}(X^{\mathsf{T}}X)\beta &= (X^{\mathsf{T}}X)^{-1}X^{\mathsf{T}}\mathbf{y}
\end{aligned}
$$

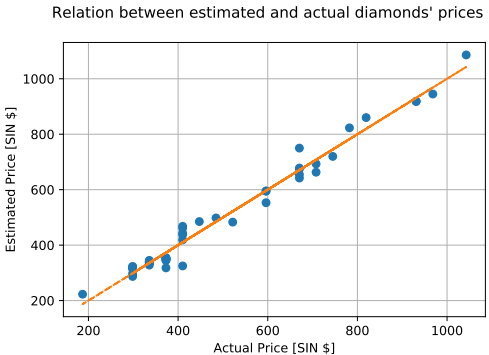
which is equivalent to the *Normal equations*

$$
\beta = (X^{\mathsf{T}}X)^{-1}X^{\mathsf{T}}\mathbf{y} \tag{1}
$$

Note that everything on the right is a set of operations on the data.

For more info, and an alternative construction of the Normal equations, see `https://goo.gl/TbLru3`.

Simple Linear Regression: Balanced Errors



What makes this look like a good fit?
The fitted line passes through the data centroid and errors pass are balanced - cf. see-saw

- More generally: a weighted sum of the errors should be 0.
- Weights should depend on the features.
- The $X^{\mathsf{T}}\epsilon = 0$ criterion works well, so we apply it.
- If you imagine the centroid as being the fulcrum of the line, viewed as a lever, we wish to "balance" the errors around that point

Simple Linear Regression: Implementation

When implemented in software, the Normal equations are not used directly: faster and more numerically accurate algorithms are used instead, but the results are equivalent in exact arithmetic (remember: digital computers perform finite-precision arithmetic and so cannot be exact).

Simple Linear Regression: Implementation

When implemented in software, the Normal equations are not used directly: faster and more numerically accurate algorithms are used instead, but the results are equivalent in exact arithmetic (remember: digital computers perform finite-precision arithmetic and so cannot be exact).

One option is to use statsmodels: consistent with R (separate model specification), excellent diagnostics as standard

Another option is to use sklearn: consistent with other sklearn algorithms, more controls

Simple Linear Regression: Implementation

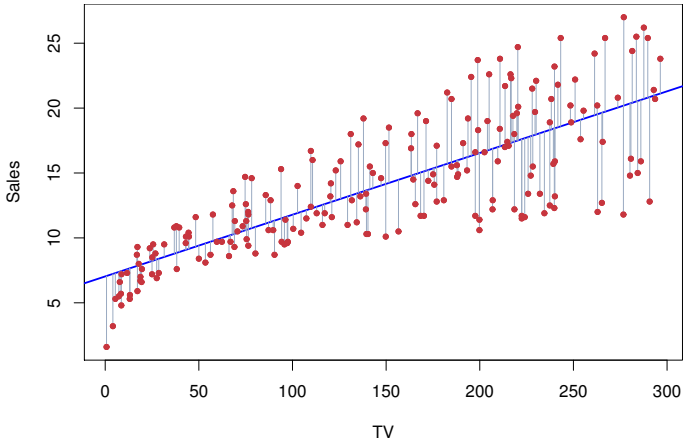
When implemented in software, the Normal equations are not used directly: faster and more numerically accurate algorithms are used instead, but the results are equivalent in exact arithmetic (remember: digital computers perform finite-precision arithmetic and so cannot be exact).

One option is to use statsmodels: consistent with R (separate model specification), excellent diagnostics as standard

Another option is to use sklearn: consistent with other sklearn algorithms, more controls

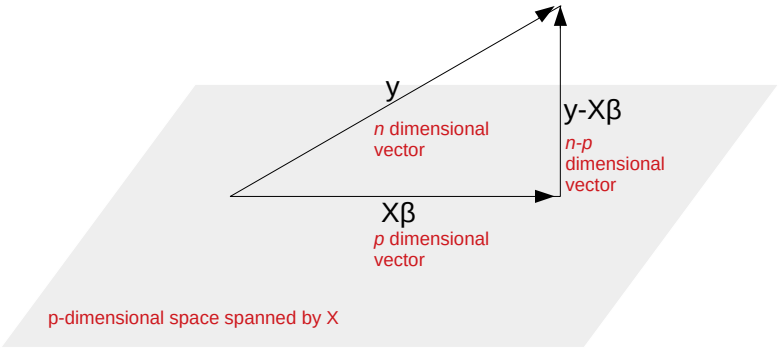
Remember: after *learning* the β parameters using the training data $\{\mathbf{x}_i, y_i\}$, with the model encoded in the feature matrix X, it is then possible to predict $\hat{y}_k$ for "new" (test) $\mathbf{x}_k$ values, using separate *prediction* function calls.

SLR: Residual Plot for the model



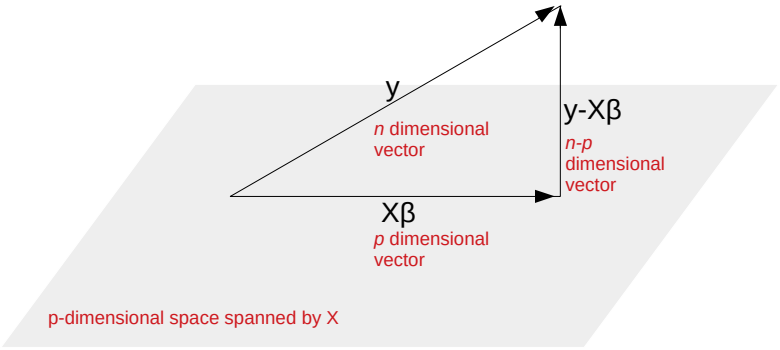
Source: ISLR, Fig 3.1: Advertising data with the model "Sales ∼ TV".
Note the vertical distance between the red dots (data points) **y** and the corresponding $\hat{\mathbf{y}}$ on the regression line, which is termed the *error* ϵ.

Geometrical interpretation of regression: n rows, p features, $n > p$



- Analogy: achieving photorealism with a limited palette of colours.
- Grey plane represents all the colours mixable from those colours.
- Point above plane: a colour that needs to be approximated.

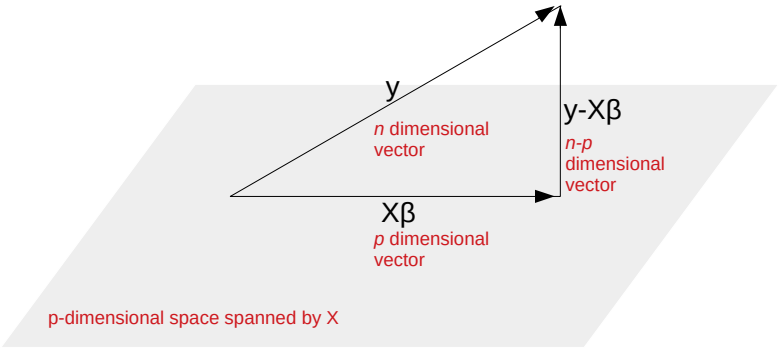
Geometrical interpretation of regression: n rows, p features, $n > p$



- Analogy: achieving photorealism with a limited palette of colours.
- Grey plane represents all the colours mixable from those colours.
- Point above plane: a colour that needs to be approximated.

- The X matrix spans the $p \times p$ space represented by the grey plane.
- But $\mathbf{y}$ has $n > p$ dimensions and so is represented by a point that lies outside the grey plane.
- When $\mathbf{y}$ is projected onto the nearest point in the X space,
  - The projected point is $\hat{\mathbf{y}}$.
  - The residuals (errors) ϵ are $\mathbf{y} - X\beta \equiv \mathbf{y} - \hat{\mathbf{y}}$.

Geometrical interpretation of regression: n rows, p features, $n > p$



- Analogy: achieving photorealism with a limited palette of colours.
- Grey plane represents all the colours mixable from those colours.
- Point above plane: a colour that needs to be approximated.

- The X matrix spans the $p \times p$ space represented by the grey plane.
- But $\mathbf{y}$ has $n > p$ dimensions and so is represented by a point that lies outside the grey plane.
- When $\mathbf{y}$ is projected onto the nearest point in the X space,
  - The projected point is $\hat{\mathbf{y}}$.
  - The residuals (errors) ϵ are $\mathbf{y} - X\beta \equiv \mathbf{y} - \hat{\mathbf{y}}$.

This decomposition of n data dimensions (observations) into p model parameters and n residuals with rank $n - p$ is helpful when interpreting regression diagnostics.

OLS and Linear Regression

Definition 3 (BLUE)

According to the Gauss-Markov theorem, *Ordinary Least Squares* (OLS), which uses the Normal equations to minimise the sum of the squares of the errors ($\|\epsilon\|_2 \equiv \sqrt{e_1^2 + e_2^2 + e_3^2 + \ldots + e_n^2}$), is the *Best, Linear, Unbiased, Estimator* of that model that can be derived from the training data, provided some reasonably loose assumptions hold.

OLS and Linear Regression

Definition 3 (BLUE)

According to the Gauss-Markov theorem, *Ordinary Least Squares* (OLS), which uses the Normal equations to minimise the sum of the squares of the errors ($\|\epsilon\|_2 \equiv \sqrt{e_1^2 + e_2^2 + e_3^2 + \ldots + e_n^2}$), is the *Best, Linear, Unbiased, Estimator* of that model that can be derived from the training data, provided some reasonably loose assumptions hold.

When we discuss Bias, Variance and Irreducible Error, it is clear that low bias is not enough. OLS might be BLUE but that does not guarantee low variance, because overfitting can still be a problem.

OLS and Linear Regression

Definition 3 (BLUE)

According to the Gauss-Markov theorem, *Ordinary Least Squares* (OLS), which uses the Normal equations to minimise the sum of the squares of the errors ($\|\epsilon\|_2 \equiv \sqrt{e_1^2 + e_2^2 + e_3^2 + \ldots + e_n^2}$), is the *Best, Linear, Unbiased, Estimator* of that model that can be derived from the training data, provided some reasonably loose assumptions hold.

When we discuss Bias, Variance and Irreducible Error, it is clear that low bias is not enough. OLS might be BLUE but that does not guarantee low variance, because overfitting can still be a problem.
In practice, the assumptions required for OLS to be appropriate can be stated in terms of properties of the residual vector ϵ.

OLS and Linear Regression

Definition 3 (BLUE)

According to the Gauss-Markov theorem, *Ordinary Least Squares* (OLS), which uses the Normal equations to minimise the sum of the squares of the errors ($\|\epsilon\|_2 \equiv \sqrt{e_1^2 + e_2^2 + e_3^2 + \ldots + e_n^2}$), is the *Best, Linear, Unbiased, Estimator* of that model that can be derived from the training data, provided some reasonably loose assumptions hold.

When we discuss Bias, Variance and Irreducible Error, it is clear that low bias is not enough. OLS might be BLUE but that does not guarantee low variance, because overfitting can still be a problem.
In practice, the assumptions required for OLS to be appropriate can be stated in terms of properties of the residual vector ϵ.
In the rest of this lecture, we will generalise from Simple to Multiple Linear Regression, where $\beta = (\beta_0, \beta_1, \beta_2, \ldots, \beta_p)$ and $2 \leq p \leq n$, so instead of fitting lines, we fit (hyper)planes to data.

Assumptions required for the linear model to be meaningful

Definition 4 (Linear Regression Assumptions)

1. The underlying relationship between the predictors and the response is linear in the regression parameters β.

Assumptions required for the linear model to be meaningful

Definition 4 (Linear Regression Assumptions)

1. The underlying relationship between the predictors and the response is linear in the regression parameters β.
2. The residual errors ϵ are drawn from a (multivariate) Normal distribution $N(\mu, \sigma^2)$ where $\mu = \mathbf{0}$.

Assumptions required for the linear model to be meaningful

Definition 4 (Linear Regression Assumptions)

1. The underlying relationship between the predictors and the response is linear in the regression parameters β.
2. The residual errors ϵ are drawn from a (multivariate) Normal distribution $N(\mu, \sigma^2)$ where $\mu = \mathbf{0}$.
3. The predictors are not pairwise collinear, i.e., each pair of predictors β_{j_1} and β_{j_2} (associated with columns $X(:, j_1)$ and $X(:, j_2)$) have low correlation (equivalently, the inner product of $X(:, j_1)$ and $X(:, j_2)$ is far from zero).

Assumptions required for the linear model to be meaningful

Definition 4 (Linear Regression Assumptions)

1. The underlying relationship between the predictors and the response is linear in the regression parameters β.
2. The residual errors ϵ are drawn from a (multivariate) Normal distribution $N(\mu, \sigma^2)$ where $\mu = \mathbf{0}$.
3. The predictors are not pairwise collinear, i.e., each pair of predictors β_{j_1} and β_{j_2} (associated with columns $X(:, j_1)$ and $X(:, j_2)$) have low correlation (equivalently, the inner product of $X(:, j_1)$ and $X(:, j_2)$ is far from zero).
4. There is no auto-correlation in $\mathbf{y}$: each observation is independent of its "neighbours".

Assumptions required for the linear model to be meaningful

Definition 4 (Linear Regression Assumptions)

1. The underlying relationship between the predictors and the response is linear in the regression parameters β.
2. The residual errors ϵ are drawn from a (multivariate) Normal distribution $N(\mu, \sigma^2)$ where $\mu = \mathbf{0}$.
3. The predictors are not pairwise collinear, i.e., each pair of predictors β_{j_1} and β_{j_2} (associated with columns $X(:, j_1)$ and $X(:, j_2)$) have low correlation (equivalently, the inner product of $X(:, j_1)$ and $X(:, j_2)$ is far from zero).
4. There is no auto-correlation in $\mathbf{y}$: each observation is independent of its "neighbours".
5. The errors are *homoscedastic* (i.e., Var(ϵ) is constant over the range of $\mathbf{x}$ or $\mathbf{y}$).

Assumptions required for the linear model to be meaningful

Definition 4 (Linear Regression Assumptions)

1. The underlying relationship between the predictors and the response is linear in the regression parameters β.
2. The residual errors ϵ are drawn from a (multivariate) Normal distribution $N(\mu, \sigma^2)$ where $\mu = \mathbf{0}$.
3. The predictors are not pairwise collinear, i.e., each pair of predictors β_{j_1} and β_{j_2} (associated with columns $X(:, j_1)$ and $X(:, j_2)$) have low correlation (equivalently, the inner product of $X(:, j_1)$ and $X(:, j_2)$ is far from zero).
4. There is no auto-correlation in $\mathbf{y}$: each observation is independent of its "neighbours".
5. The errors are *homoscedastic* (i.e., Var(ϵ) is constant over the range of $\mathbf{x}$ or $\mathbf{y}$).

Because these assumptions depend both on the data and on the model fitted to that data, it is meaningless to say that "Data set A does not satisfy the linear regression assumptions", because this observation might not apply to all formulations of all models applied to that data.

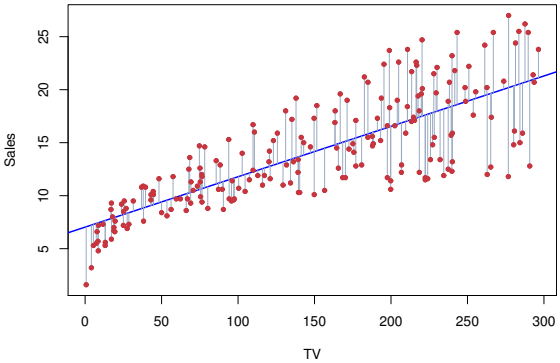
Assumptions required for the linear model to be meaningful

Definition 4 (Linear Regression Assumptions)

1. The underlying relationship between the predictors and the response is linear in the regression parameters β.
2. The residual errors ϵ are drawn from a (multivariate) Normal distribution $N(\mu, \sigma^2)$ where $\mu = \mathbf{0}$.
3. The predictors are not pairwise collinear, i.e., each pair of predictors β_{j_1} and β_{j_2} (associated with columns $X(:, j_1)$ and $X(:, j_2)$) have low correlation (equivalently, the inner product of $X(:, j_1)$ and $X(:, j_2)$ is far from zero).
4. There is no auto-correlation in $\mathbf{y}$: each observation is independent of its "neighbours".
5. The errors are *homoscedastic* (i.e., Var(ϵ) is constant over the range of $\mathbf{x}$ or $\mathbf{y}$).

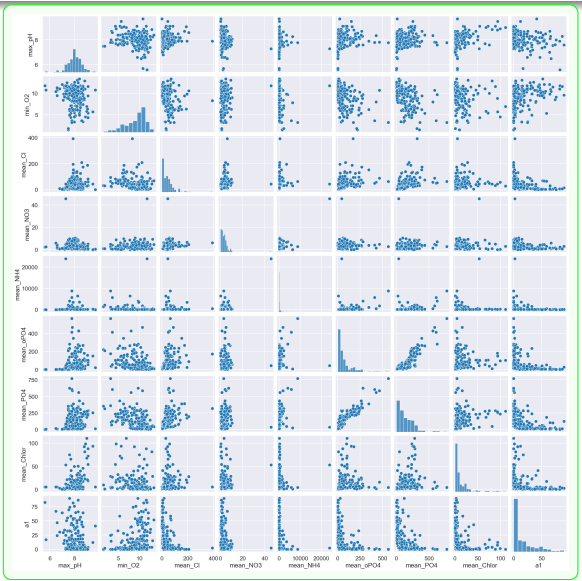
Because these assumptions depend both on the data and on the model fitted to that data, it is meaningless to say that "Data set A does not satisfy the linear regression assumptions", because this observation might not apply to all formulations of all models applied to that data.
Consequently, these assumptions can be used constructively, when model building, or as checks, when validating models.

Linear relationship



- In both cases, the relationship between predictor (feature) and target is approximately linear.
- Given a feature value, we can predict the target value using a simple linear formula.
- The predicted parameters are the *intercept* β_0 and *slope* β_1 of the line.
- Usually the vertical distance between a data point x_i and its predicted value $\hat{y}_i$ is $\epsilon_i \neq 0$.
- ϵ_i is the *residual error*. It quantifies data behaviour not included in our model.

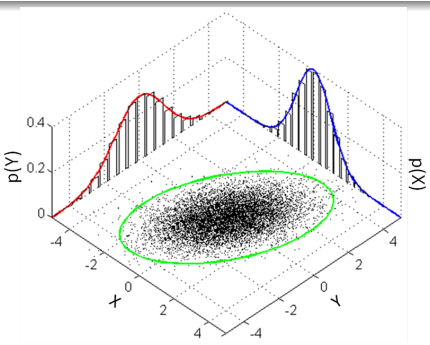
Collinearity (high pairwise correlation) among the algae bloom predictors



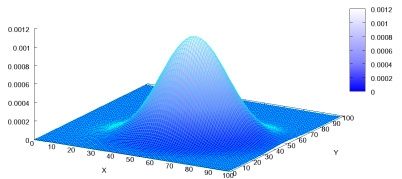
- The pairplot confirms what we saw in the corresponding correlation matrix: `mean_PO4` and `mean_OPO4` are highly correlated with each other (indeed, the relevant scatterplots indicate a strong linear relationship).
- Either `mean_PO4` or `mean_OPO4` can be included in the model, but not both of them.
- Also, the individual predictors do not have a strong linear relationship with `a1` (look at the scatterplots in the last row and column) so, on their own, they are not likely to predict `a1` well with a linear model.
- However, it is still possible that a combination of predictors might predict a1 well.

Errors are normally distributed

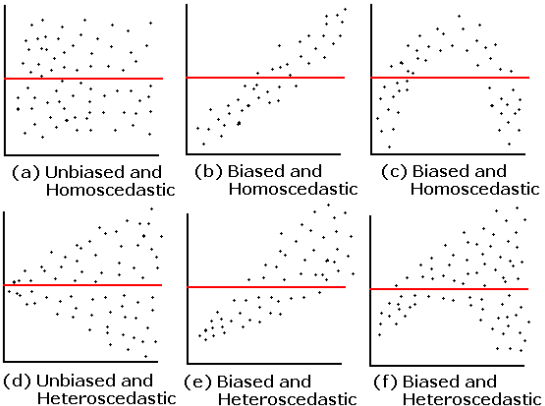
- Centred on zero so small errors are more common
- Symmetric so positive and negative balance out
- Normal distribution is a also called the Gaussian distribution and is “bell-shaped”.



Multivariate Normal Distribution



Bias and variance in regression

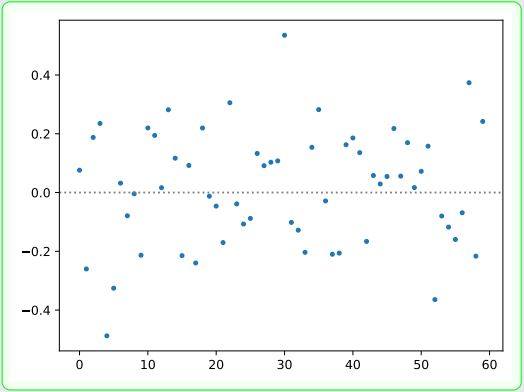


Source: https://bit.ly/3vC9zK7

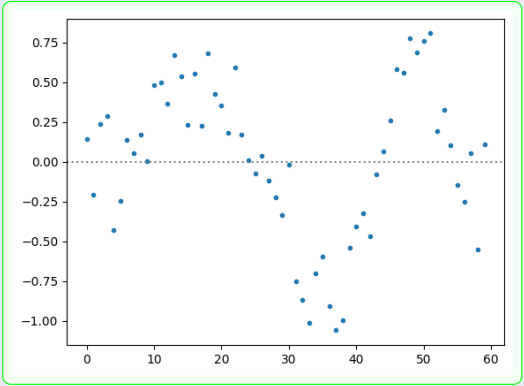
- Bias is caused by underfitting.
- Fix bias by adding suitable predictors.
- Overfitting causes large variance.
- If variance changes over the range, some errors get undue attention.
- Fix this by weighting the errors so they are balanced.

Errors should not be serially correlated

No serial correlation



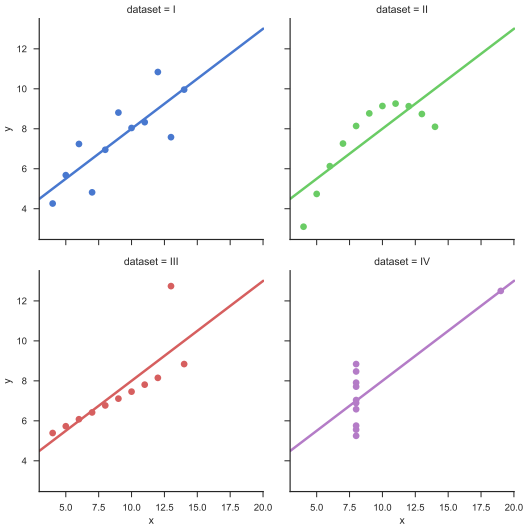
Positive serial correlation



Apparent seasonal effects - can they be removed?

1. Add feature to the model
2. Include autoregressive terms (but then it is no longer Ordinary Least Squares (OLS)!)

Anscombe's quartet (1973)

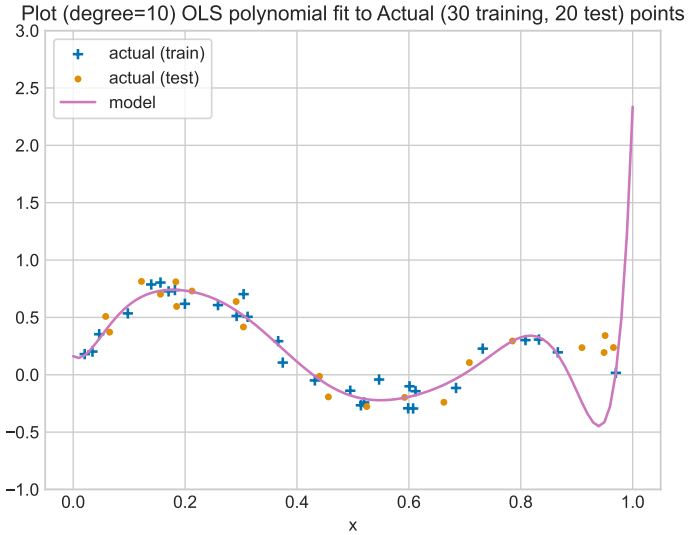


Francis Anscombe devised 4 data sets to show different forms of misalignment between data and models. Sets I,II,III share the same x values. All 4 sets share approximately the same descriptive statistics (mean and variance), but little else is common to all 4!

Only I appears suited as it stands. The other data sets require some work, particularly IV.

What do you think needs to be done for each data set?

What's happening here???



Common Cost Functions in Regression Models

Remember: we are trying to minimise a loss function based on the error, which we approximate with the residuals of the training set.

Measure	Definition	Purpose
Mean square error (MSE)	$\frac{(p_1-a_1)^2+\cdots+(p_m-a_m)^2}{m}$	Mathematically tractable but places greater emphasise on observations with large error
Root mean square error (RMSE)	$\sqrt{\frac{(p_1-a_1)^2+\cdots+(p_m-a_m)^2}{m}}$	Has same units as data
Mean absolute error (RMAE)	$\frac{\lvert p_1-a_1\rvert+\cdots+\lvert p_m-a_m\rvert}{m}$	Does not overemphasise observations with large error (like MSE does)
Relative square error (RSE)	$\frac{(p_1-a_1)^2+\cdots+(p_m-a_m)^2}{(p_1-\bar{a})^2+\cdots+(p_m-\bar{a})^2}$	Relative metric compares the error in the predictions with errors in the simplest model possible (a model just always predicting the average value of y)
Root Relative square error (RRSE)	$\sqrt{\frac{(p_1-a_1)^2+\cdots+(p_m-a_m)^2}{(p_1-\bar{a})^2+\cdots+(p_m-\bar{a})^2}}$	
Relative absolute error (RAE)	$\frac{\lvert p_1-a_1\rvert+\cdots+\lvert p_m-a_m\rvert}{\lvert p_1-\bar{a}\rvert+\cdots+\lvert p_m-\bar{a}\rvert}$	

where a_j is the actual value, p_j is the predicted value, m is the number of observations, and $\bar{a}$ represents the mean of the a_j.

Choices of Vector norms

Definition 5 (Manhattan norm)

$\ell_1(\ldots) = \|\ldots\|_1$ is the *Manhattan* norm (length) of a vector. Let $\mathbf{x} = (x_1, x_2, \ldots, x_m)$. Then $\ell_1(\ldots) = \|\ldots\|_1 = |x_1| + |x_2| + \ldots + |x_m|$ is the *Manhattan* distance of $\mathbf{x}$ from the origin. Think of having to *walk* from one junction in Manhattan to another, the distance is the difference in the Street numbers plus the difference in the Avenue numbers.

Choices of Vector norms

Definition 5 (Manhattan norm)

$\ell_1(\ldots) = \|\ldots\|_1$ is the *Manhattan* norm (length) of a vector. Let $\mathbf{x} = (x_1, x_2, \ldots, x_m)$. Then $\ell_1(\ldots) = \|\ldots\|_1 = |x_1| + |x_2| + \ldots + |x_m|$ is the *Manhattan* distance of $\mathbf{x}$ from the origin. Think of having to *walk* from one junction in Manhattan to another, the distance is the difference in the Street numbers plus the difference in the Avenue numbers.

Definition 6 (Euclidean norm)

$\ell_2(\ldots) = \|\ldots\|_2$ is the *Euclidean* norm (length) of a vector. Let $\mathbf{x} = (x_1, x_2, \ldots, x_m)$. Then $\ell_2(\ldots) = \|\ldots\|_2 = \sqrt{x_1^2 + x_2^2 + \ldots + x_m^2}$ is the *Euclidean* distance of $\mathbf{x}$ from the origin. Think of being able to *fly* over all the buildings using the shortest route (think: Pythagoras theorem!) from one junction in Manhattan to another.

Choices of Vector norms

Definition 5 (Manhattan norm)

$\ell_1(\ldots) = \|\ldots\|_1$ is the *Manhattan* norm (length) of a vector. Let $\mathbf{x} = (x_1, x_2, \ldots, x_m)$. Then $\ell_1(\ldots) = \|\ldots\|_1 = |x_1| + |x_2| + \ldots + |x_m|$ is the *Manhattan* distance of $\mathbf{x}$ from the origin. Think of having to *walk* from one junction in Manhattan to another, the distance is the difference in the Street numbers plus the difference in the Avenue numbers.

Definition 6 (Euclidean norm)

$\ell_2(\ldots) = \|\ldots\|_2$ is the *Euclidean* norm (length) of a vector. Let $\mathbf{x} = (x_1, x_2, \ldots, x_m)$. Then $\ell_2(\ldots) = \|\ldots\|_2 = \sqrt{x_1^2 + x_2^2 + \ldots + x_m^2}$ is the *Euclidean* distance of $\mathbf{x}$ from the origin. Think of being able to *fly* over all the buildings using the shortest route (think: Pythagoras theorem!) from one junction in Manhattan to another.

The Euclidean norm is very common, but the Manhattan norm is gaining popularity, because it is robust to outliers and computers are becoming powerful enough. However we generally use Euclidean norm in this module.

Sidebar: Distance Measures for numeric data

Definition 7 (Minkowski $p-$norm)

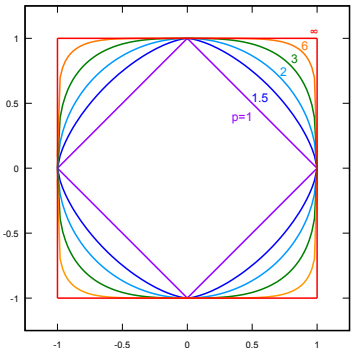
For a real number $1 \leq p < \infty$, the p-norm of $\mathbf{x}$ is defined by

$$\|\mathbf{x}\|_p \equiv \left(|x_1|^p + |x_2|^p + \ldots + |x_n|^p\right)^{\frac{1}{p}}.$$

The limiting case of $p = \infty$ is defined as

$$\|\mathbf{x}\|_\infty \equiv \max\{|x_1|, |x_2|, \ldots, |x_n|\}.$$

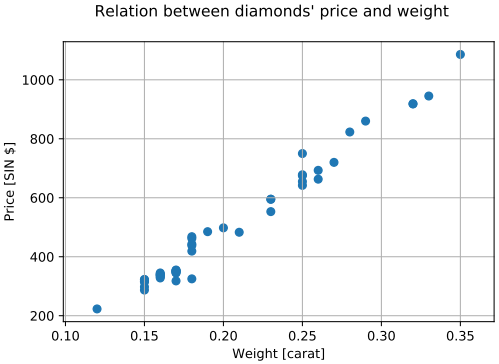
See the visualisation of the "unit balls" alongside, for $p = 1, 1.5, 2, 3, 6, \infty$.

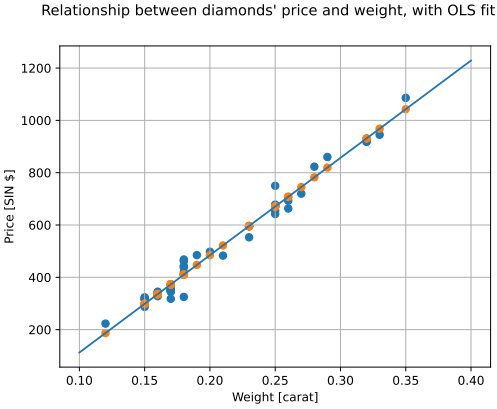


Source: wikipedia

The most common norms are when $p = 1, 2, \text{or}, \infty$. Choice of p depends on the application scenario. Can you think of when you would use each?

Case Study 2: Diamonds - Check relationship



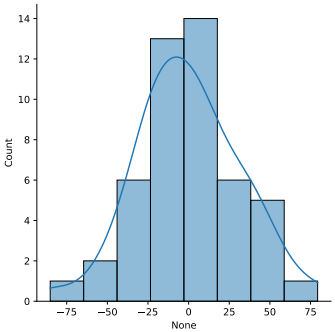


Clearly there is a linear relationship between a diamond's weight (in carats) and its price (in Singapore dollars, as here). So that is one assumption satisfied!

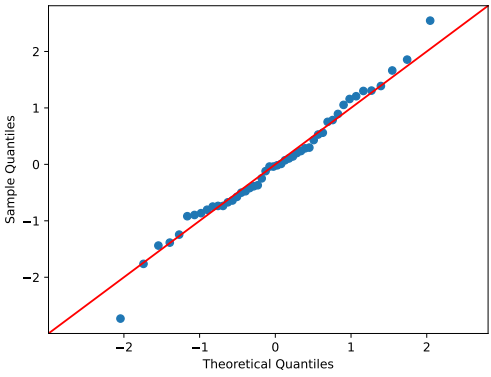
Sometimes the dependent variable has a linear dependence on a (computed) feature that is a function of one or more feature columns in the data. Example functions include log, exp, sqrt, polynomial, etc. Even if the function is nonlinear in the feature x (e.g., x^2, $\sqrt{x}$ or $\log(x)$), that does not matter, as long as the model is linear in the regression parameters β.

Case Study 2: Diamonds - Check residual distribution

1
```
import seaborn as sns
resFig = "res/residHist.pdf"
sns_plot = sns.displot(x = residuals, kde=True)
sns_plot.savefig(resFig)
```

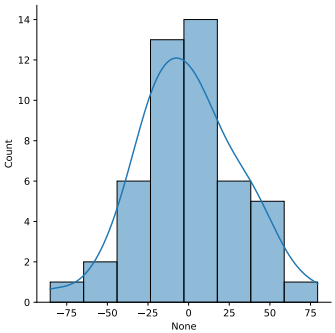


2
```
# Q-Q plot to verify the residuals distribution
resFig = "res/residualsqq.pdf"
fig = sm.qqplot(residuals, fit=True, line = '45')
fig.savefig(resFig)
```

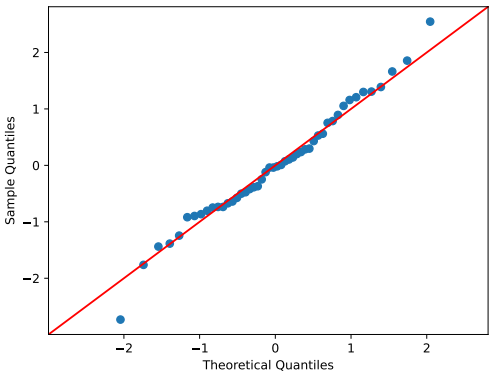


Case Study 2: Diamonds - Check residual distribution

```
import seaborn as sns
resFig = "res/residHist.pdf"
sns_plot = sns.displot(x = residuals, kde=True)
sns_plot.savefig(resFig)
```



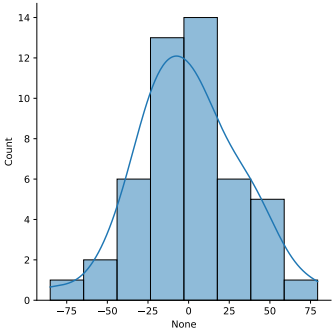

```
# Q-Q plot to verify the residuals distribution
resFig = "res/residualsqq.pdf"
fig = sm.qqplot(residuals, fit=True, line = '45')
fig.savefig(resFig)
```



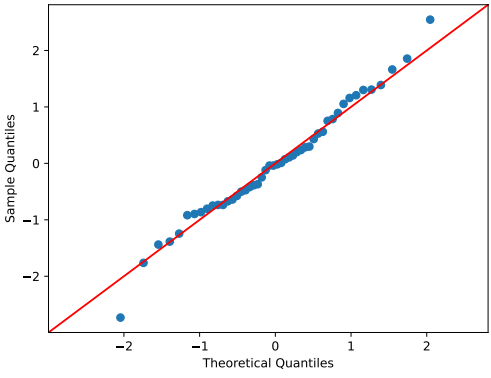
Both diagnostic plots indicate the residuals are reasonably close to Normal distribution centred on 0. Looking good so far!

Case Study 2: Diamonds - Check residual distribution

```
import seaborn as sns
resFig = "res/residHist.pdf"
sns_plot = sns.displot(x = residuals, kde=True)
sns_plot.savefig(resFig)
```




```
# Q-Q plot to verify the residuals distribution
resFig = "res/residualsqq.pdf"
fig = sm.qqplot(residuals, fit=True, line = '45')
fig.savefig(resFig)
```



Both diagnostic plots indicate the residuals are reasonably close to Normal distribution centred on 0. Looking good so far!

Is the standardised residual distribution heavy-tailed or light-tailed relative to the Normal distribution? Any other features?

Case Study 2: Diamonds - model summary

```
7 simpleModel.summary()
```

Dep. Variable:	price	**R-squared:**	0.978
Model:	OLS	**Adj. R-squared:**	0.978
Method:	Least Squares	**F-statistic:**	2070.
Date:		**Prob (F-statistic):**	6.75e-40
Time:		**Log-Likelihood:**	-233.20
No. Observations:	48	**AIC:**	470.4
Df Residuals:	46	**BIC:**	474.1
Df Model:	1		
Covariance Type:	nonrobust		

	coef	std err	t	P>|t|	[0.025	0.975]
const	-259.6259	17.319	-14.991	0.000	-294.487	-224.765
carats	3721.0249	81.786	45.497	0.000	3556.398	3885.651

Omnibus:	0.739	**Durbin-Watson:**	1.994
Prob(Omnibus):	0.691	**Jarque-Bera (JB):**	0.181
Skew:	0.056	**Prob(JB):**	0.913
Kurtosis:	3.280	**Cond. No.**	18.5

Case Study 2: Diamonds - model summary

Dep. Variable:	price	**R-squared:**	0.978
Model:	OLS	**Adj. R-squared:**	0.978
Method:	Least Squares	**F-statistic:**	2070.
Date:		**Prob (F-statistic):**	6.75e-40
Time:		**Log-Likelihood:**	-233.20
No. Observations:	48	**AIC:**	470.4
Df Residuals:	46	**BIC:**	474.1
Df Model:	1		
Covariance Type:	nonrobust		

	coef	std err	t	P>|t|	[0.025	0.975]
const	-259.6259	17.319	-14.991	0.000	-294.487	-224.765
carats	3721.0249	81.786	45.497	0.000	3556.398	3885.651

Omnibus:	0.739	**Durbin-Watson:**	1.994
Prob(Omnibus):	0.691	**Jarque-Bera (JB):**	0.181
Skew:	0.056	**Prob(JB):**	0.913
Kurtosis:	3.280	**Cond. No.**	18.5

```
simpleModel.summary()
```

1. **How much of the variability of the data is explained by the model?**

Case Study 2: Diamonds - model summary

Dep. Variable:	price	**R-squared:**	0.978
Model:	OLS	**Adj. R-squared:**	0.978
Method:	Least Squares	**F-statistic:**	2070.
Date:		**Prob (F-statistic):**	6.75e-40
Time:		**Log-Likelihood:**	-233.20
No. Observations:	48	**AIC:**	470.4
Df Residuals:	46	**BIC:**	474.1
Df Model:	1		
Covariance Type:	nonrobust		

	coef	std err	t	P>|t|	[0.025	0.975]
const	-259.6259	17.319	-14.991	0.000	-294.487	-224.765
carats	3721.0249	81.786	45.497	0.000	3556.398	3885.651

Omnibus:	0.739	**Durbin-Watson:**	1.994
Prob(Omnibus):	0.691	**Jarque-Bera (JB):**	0.181
Skew:	0.056	**Prob(JB):**	0.913
Kurtosis:	3.280	**Cond. No.**	18.5

9
```
simpleModel.summary()
```

1. **How much of the variability of the data is explained by the model?**
2. **What is the probability that such data arose if price does not increase with weight?**

Case Study 2: Diamonds - model summary

Dep. Variable:	price	**R-squared:**	0.978
Model:	OLS	**Adj. R-squared:**	0.978
Method:	Least Squares	**F-statistic:**	2070.
Date:		**Prob (F-statistic):**	6.75e-40
Time:		**Log-Likelihood:**	-233.20
No. Observations:	48	**AIC:**	470.4
Df Residuals:	46	**BIC:**	474.1
Df Model:	1		
Covariance Type:	nonrobust		

	coef	std err	t	P>|t|	[0.025	0.975]
const	-259.6259	17.319	-14.991	0.000	-294.487	-224.765
carats	3721.0249	81.786	45.497	0.000	3556.398	3885.651

Omnibus:	0.739	**Durbin-Watson:**	1.994
Prob(Omnibus):	0.691	**Jarque-Bera (JB):**	0.181
Skew:	0.056	**Prob(JB):**	0.913
Kurtosis:	3.280	**Cond. No.**	18.5

```
10 simpleModel.summary()
```

1. **How much of the variability of the data is explained by the model?**
2. **What is the probability that such data arose if price does not increase with weight?**
3. **What score(s) indicate that the distribution of the residuals is Normal?**

Case Study 2: Diamonds - model summary

Dep. Variable:	price	**R-squared:**	0.978
Model:	OLS	**Adj. R-squared:**	0.978
Method:	Least Squares	**F-statistic:**	2070.
Date:		**Prob (F-statistic):**	6.75e-40
Time:		**Log-Likelihood:**	-233.20
No. Observations:	48	**AIC:**	470.4
Df Residuals:	46	**BIC:**	474.1
Df Model:	1		
Covariance Type:	nonrobust		

	coef	std err	t	P>|t|	[0.025	0.975]
const	-259.6259	17.319	-14.991	0.000	-294.487	-224.765
carats	3721.0249	81.786	45.497	0.000	3556.398	3885.651

Omnibus:	0.739	**Durbin-Watson:**	1.994
Prob(Omnibus):	0.691	**Jarque-Bera (JB):**	0.181
Skew:	0.056	**Prob(JB):**	0.913
Kurtosis:	3.280	**Cond. No.**	18.5

11
```
simpleModel.summary()
```

1. **How much of the variability of the data is explained by the model?**
2. **What is the probability that such data arose if price does not increase with weight?**
3. **What score(s) indicate that the distribution of the residuals is Normal?**
4. **How could you compare this model with another from the same family?**

Case Study 2: Diamonds - model summary

Dep. Variable:	price	**R-squared:**	0.978
Model:	OLS	**Adj. R-squared:**	0.978
Method:	Least Squares	**F-statistic:**	2070.
Date:		**Prob (F-statistic):**	6.75e-40
Time:		**Log-Likelihood:**	-233.20
No. Observations:	48	**AIC:**	470.4
Df Residuals:	46	**BIC:**	474.1
Df Model:	1		
Covariance Type:	nonrobust		

	coef	std err	t	P>|t|	[0.025	0.975]
const	-259.6259	17.319	-14.991	0.000	-294.487	-224.765
carats	3721.0249	81.786	45.497	0.000	3556.398	3885.651

Omnibus:	0.739	**Durbin-Watson:**	1.994
Prob(Omnibus):	0.691	**Jarque-Bera (JB):**	0.181
Skew:	0.056	**Prob(JB):**	0.913
Kurtosis:	3.280	**Cond. No.**	18.5

12
```
simpleModel.summary()
```

1. **How much of the variability of the data is explained by the model?**
2. **What is the probability that such data arose if price does not increase with weight?**
3. **What score(s) indicate that the distribution of the residuals is Normal?**
4. **How could you compare this model with another from the same family?**
5. **Which metric measures whether there is autocorrelation in y?**

Case Study 2: Diamonds - model summary

Dep. Variable:	price	**R-squared:**	0.978
Model:	OLS	**Adj. R-squared:**	0.978
Method:	Least Squares	**F-statistic:**	2070.
Date:		**Prob (F-statistic):**	6.75e-40
Time:		**Log-Likelihood:**	-233.20
No. Observations:	48	**AIC:**	470.4
Df Residuals:	46	**BIC:**	474.1
Df Model:	1		
Covariance Type:	nonrobust		

	coef	std err	t	P>|t|	[0.025	0.975]
const	-259.6259	17.319	-14.991	0.000	-294.487	-224.765
carats	3721.0249	81.786	45.497	0.000	3556.398	3885.651

Omnibus:	0.739	**Durbin-Watson:**	1.994
Prob(Omnibus):	0.691	**Jarque-Bera (JB):**	0.181
Skew:	0.056	**Prob(JB):**	0.913
Kurtosis:	3.280	**Cond. No.**	18.5

13
```
simpleModel.summary()
```

1. **How much of the variability of the data is explained by the model?**
2. **What is the probability that such data arose if price does not increase with weight?**
3. **What score(s) indicate that the distribution of the residuals is Normal?**
4. **How could you compare this model with another from the same family?**
5. **Which metric measures whether there is autocorrelation in y?**
6. **Which metric measures overall feature-feature correlation?**

Model summary interpretation - 1

OLS Regression Results

Dep. Variable:	price	**R-squared:**	0.978
Model:	OLS	**Adj. R-squared:**	0.978
Method:	Least Squares	**F-statistic:**	2070.
Date:	Sun, 16 Feb 2025	**Prob (F-statistic):**	6.75e-40
Time:	12:13:31	**Log-Likelihood:**	-233.20
No. Observations:	48	**AIC:**	470.4
Df Residuals:	46	**BIC:**	474.1
Df Model:	1		
Covariance Type:	nonrobust		

	coef	std err	t	P>|t|	[0.025	0.975]
const	-259.6259	17.319	-14.991	0.000	-294.487	-224.765
carats	3721.0249	81.786	45.497	0.000	3556.398	3885.651

Omnibus:	0.739	**Durbin-Watson:**	1.994
Prob(Omnibus):	0.691	**Jarque-Bera (JB):**	0.181
Skew:	0.056	**Prob(JB):**	0.913
Kurtosis:	3.280	**Cond. No.**	18.5

Notes:
[1] Standard Errors assume that the covariance matrix of the errors is correctly specified.

Definition 8 (Dep. variable)

This is synonymous with the *target*, which is `price` (in this dataset).

Definition 9 (Model)

statsmodels uses it here in the sense of *problem formulation*. We wish to solve an Ordinary Least Squares problem (assumes all the regression assumptions are met, so no special treatment was applied).

Definition 10 (No. Observations)

This is the number of rows (also known as instances or cases) in the training set.

Model summary interpretation - 2

Definition 11 (Df Model)

The model has one named feature (carats (weight of the diamond)) and one unnamed feature (constant, independent of carats). df, the number of degrees of freedom counts the named features.

Definition 12 (Df Residuals)

The number of degrees of freedom in the residuals is the number of residuals minus the number of features. A higher value tends to go with smaller model variance.

Definition 13 (Covariance Type)

If residuals have the same variance (homoscedastic), nonrobust covariance (the default) can be used.

OLS Regression Results

Dep. Variable:	price	**R-squared:**	0.978
Model:	OLS	**Adj. R-squared:**	0.978
Method:	Least Squares	**F-statistic:**	2070.
Date:	Sun, 16 Feb 2025	**Prob (F-statistic):**	6.75e-40
Time:	12:13:31	**Log-Likelihood:**	-233.20
No. Observations:	48	**AIC:**	470.4
Df Residuals:	46	**BIC:**	474.1
Df Model:	1		
Covariance Type:	nonrobust		

	coef	std err	t	P>|t|	[0.025	0.975]
const	-259.6259	17.319	-14.991	0.000	-294.487	-224.765
carats	3721.0249	81.786	45.497	0.000	3556.398	3885.651

Omnibus:	0.739	**Durbin-Watson:**	1.994
Prob(Omnibus):	0.691	**Jarque-Bera (JB):**	0.181
Skew:	0.056	**Prob(JB):**	0.913
Kurtosis:	3.280	**Cond. No.**	18.5

Notes:
[1] Standard Errors assume that the covariance matrix of the errors is correctly specified.

Model summary interpretation - 3

OLS Regression Results

Dep. Variable:	price	**R-squared:**	0.978
Model:	OLS	**Adj. R-squared:**	0.978
Method:	Least Squares	**F-statistic:**	2070.
Date:	Sun, 16 Feb 2025	**Prob (F-statistic):**	6.75e-40
Time:	12:13:31	**Log-Likelihood:**	-233.20
No. Observations:	48	**AIC:**	470.4
Df Residuals:	46	**BIC:**	474.1
Df Model:	1		
Covariance Type:	nonrobust		

	coef	std err	t	P>|t|	[0.025	0.975]
const	-259.6259	17.319	-14.991	0.000	-294.487	-224.765
carats	3721.0249	81.786	45.497	0.000	3556.398	3885.651

Omnibus:	0.739	**Durbin-Watson:**	1.994
Prob(Omnibus):	0.691	**Jarque-Bera (JB):**	0.181
Skew:	0.056	**Prob(JB):**	0.913
Kurtosis:	3.280	**Cond. No.**	18.5

Notes:
[1] Standard Errors assume that the covariance matrix of the errors is correctly specified.

Definition 14 (R-squared)

This is the ratio of the data variance explained by the model, to the variance of the data. It ranges from zero (model explains none of the data variance) to one (model explains all the data variance). A higher value is better, but be careful of overfitting the training set!

Definition 15 (Adj. R-squared)

Similar to R-squared, but it takes account of the number of features. Adding a feature generally increases R-squared, but if the feature did not help as much as its peers, adjusted R-squared shows a smaller increase than “normal” R-squared.

Model summary interpretation - 4

Definition 16 (F-statistic)

Ratio of the variance of a model with just the constant (intercept) feature to the variance of this model. Generally, large values of F are preferred.

Definition 17 (Prob (F statistic))

The value is assumed to follow the F distribution for given *dof*, so can lookup its probability. Small probability indicates that it is highly *unlikely* that the model is doing well purely by chance.

Definition 18 (log likelihood)

OLS is a special case of *maximum likelihood estimation*. Larger likelihood model fits the training data better.

OLS Regression Results

Dep. Variable:	price	**R-squared:**	0.978
Model:	OLS	**Adj. R-squared:**	0.978
Method:	Least Squares	**F-statistic:**	2070.
Date:	Sun, 16 Feb 2025	**Prob (F-statistic):**	6.75e-40
Time:	12:13:31	**Log-Likelihood:**	-233.20
No. Observations:	48	**AIC:**	470.4
Df Residuals:	46	**BIC:**	474.1
Df Model:	1		
Covariance Type:	nonrobust		

	coef	std err	t	P>|t|	[0.025	0.975]
const	-259.6259	17.319	-14.991	0.000	-294.487	-224.765
carats	3721.0249	81.786	45.497	0.000	3556.398	3885.651

Omnibus:	0.739	**Durbin-Watson:**	1.994
Prob(Omnibus):	0.691	**Jarque-Bera (JB):**	0.181
Skew:	0.056	**Prob(JB):**	0.913
Kurtosis:	3.280	**Cond. No.**	18.5

Notes:
[1] Standard Errors assume that the covariance matrix of the errors is correctly specified.

Model summary interpretation - 5

OLS Regression Results

Dep. Variable:	price	**R-squared:**	0.978
Model:	OLS	**Adj. R-squared:**	0.978
Method:	Least Squares	**F-statistic:**	2070.
Date:	Sun, 16 Feb 2025	**Prob (F-statistic):**	6.75e-40
Time:	12:13:31	**Log-Likelihood:**	-233.20
No. Observations:	48	**AIC:**	470.4
Df Residuals:	46	**BIC:**	474.1
Df Model:	1		
Covariance Type:	nonrobust		

	coef	std err	t	P>|t|	[0.025	0.975]
const	-259.6259	17.319	-14.991	0.000	-294.487	-224.765
carats	3721.0249	81.786	45.497	0.000	3556.398	3885.651

Omnibus:	0.739	**Durbin-Watson:**	1.994
Prob(Omnibus):	0.691	**Jarque-Bera (JB):**	0.181
Skew:	0.056	**Prob(JB):**	0.913
Kurtosis:	3.280	**Cond. No.**	18.5

Notes:
[1] Standard Errors assume that the covariance matrix of the errors is correctly specified.

Definition 19 (AIC and BIC)

Akaike and Bayesian Information Criterion. These are calculated from the residuals and are derived from *information theory*. They allow for the number of features. Lower values are better.

Definition 20 (Features table: const,carats in this example)

coef is the parameter value for that feature, e.g., const=-259.6 here. $P > |t| = 0$ so it is highly unlikely the coef is zero, given the training data. We also have the 2.5% and 97.5% quantiles, giving the expected range of coef.

Model summary interpretation - 6

Definition 21 (Skew, Kurtosis)

Measures of asymmetry and of peak shape of the residual distribution. Ideal values are 0 (skew) and 3 (kurtosis).

Definition 22 (Durbin-Watson)

Measures the serial correlation of the residuals. Ideal value is 2 (no serial correlation).

Definition 23 (Cond. no)

OLS implementation solves a linear system of equations. Condition number measures column (hence feature independence). Large values mean the features are not independent (they are correlated), making the system more difficult to solve.

OLS Regression Results

Dep. Variable:	price	**R-squared:**	0.978
Model:	OLS	**Adj. R-squared:**	0.978
Method:	Least Squares	**F-statistic:**	2070.
Date:	Sun, 16 Feb 2025	**Prob (F-statistic):**	6.75e-40
Time:	12:13:31	**Log-Likelihood:**	-233.20
No. Observations:	48	**AIC:**	470.4
Df Residuals:	46	**BIC:**	474.1
Df Model:	1		
Covariance Type:	nonrobust		

	coef	std err	t	P>|t|	[0.025	0.975]
const	-259.6259	17.319	-14.991	0.000	-294.487	-224.765
carats	3721.0249	81.786	45.497	0.000	3556.398	3885.651

Omnibus:	0.739	**Durbin-Watson:**	1.994
Prob(Omnibus):	0.691	**Jarque-Bera (JB):**	0.181
Skew:	0.056	**Prob(JB):**	0.913
Kurtosis:	3.280	**Cond. No.**	18.5

Notes:
[1] Standard Errors assume that the covariance matrix of the errors is correctly specified.

What does all this mean?

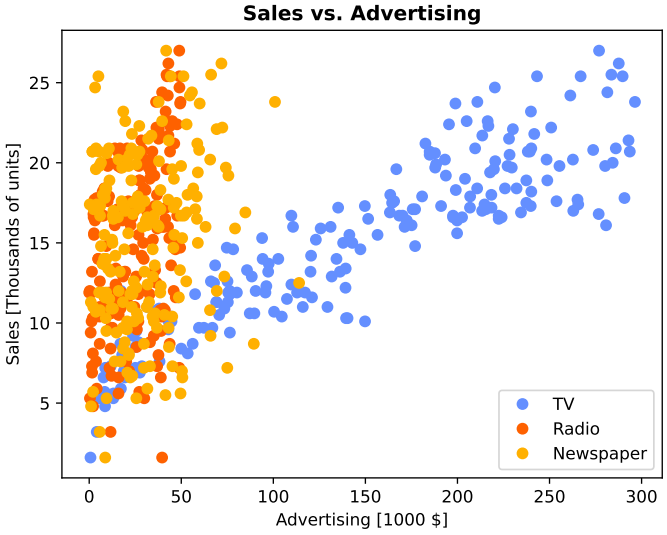
- Clearly, Linear Regression provides many diagnostics to assess how well the model works on the *training set*.
- We can test the model assumptions and many related desirable properties, when applied to the training set.
- However, in machine learning, our goal is to minimise prediction error on unseen data.
- As described last week, we need to perform a train-(validation)-test split and evaluate the errors on the test set.
- *The best model is the one that minimises the error when predicting the target in the test set!*
- The Regression Model diagnostics can help
  1. If a model does not do well, they can help to diagnose the problem(s).
  2. They can also be used constructively, to help identify promising candidate models.

Case Study 3: Advertising: Data and Hypotheses

	TV	Radio	Newspaper	Sales
0	230.1	37.8	69.2	22.1
1	44.5	39.3	45.1	10.4
2	17.2	45.9	69.3	12.0
3	151.5	41.3	58.5	16.5
4	180.8	10.8	58.4	17.9

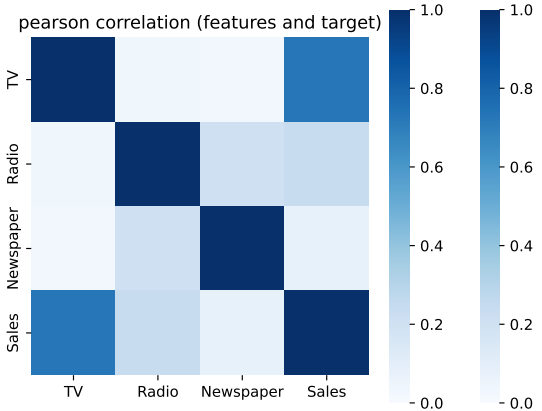
- In this data set, the sales figure captures how many thousands of widgets of a particular type were sold in a year.
- Newspaper, Radio and TV represent the annual spend per widget type on the associated advertising channel.
- The hypothesis is that spend on advertising is a good predictor of sales performance.
- Since marketing budgets are limited, where should the adverts be placed for maximum sales?
- Alternatively, how should marketing funds be distributed across the 3 channels to achieve a specified sales performance, while keeping the total spend as low as possible?

Case Study 3: Advertising: Looking at the data



Which of the advertising channels appear to have a linear relationship with Sales?

Case Study 3: Advertising: Collinearity?



- Correlation matrix can indicate which features should participate in the model as predictors.
- A good predictor should have a high correlation with the target (Sales in this case) and should have low correlation with other candidate predictors.
- **What are expected to be good predictors for this data?**
  - Sales (the target) is placed in the last row (or column).
  - TV $>$ Radio $>$ Newspaper, with moderate correlation between Radio and Newspaper.

Sidebar: specifying models

The statsmodels way

- The dataframe contains the observed variables

The sklearn way

- The dataframe contains the (computed) features

Sidebar: specifying models

The statsmodels way

- The dataframe contains the observed variables
- The model is specified separately

The sklearn way

- The dataframe contains the (computed) features
- The model is defined implicitly

Sidebar: specifying models

The statsmodels way

- The dataframe contains the observed variables
- The model is specified separately
- Easier to change the model when experimenting

The sklearn way

- The dataframe contains the (computed) features
- The model is defined implicitly
- Standard interface across all sklearn

Sidebar: specifying models

The statsmodels way

- The dataframe contains the observed variables
- The model is specified separately
- Easier to change the model when experimenting

The sklearn way

- The dataframe contains the (computed) features
- The model is defined implicitly
- Standard interface across all sklearn

- statsmodels models are expressed like "Sales $\sim$ TV * Radio + poly(Newspaper,2)". This notation came from the applied statistics community.
- In words: “Sales depends on TV spending, Radio spending, the interaction between TV and Radio spending, Newspaper spending and Newspaper spending squared (5 features from 3 measured features).”
- statsmodels offers its own plotting (like seaborn but not as good). Its model summary is very convenient.
- sklearn exposes more of the details (e.g., choice of algorithm and configuration parameters).
- Both statsmodels and sklearn use the same libraries (scipy, numpy, etc.) underneath.

Case Study 3: Advertising: Model Building (“stats” way)

- Start from a “full model” and prune, versus from an “empty model” and add
- We choose the latter, as it is often easier to avoid overfitting

Case Study 3: Advertising: Model Building ("stats" way)

- Start from a "full model" and prune, versus from an "empty model" and add
- We choose the latter, as it is often easier to avoid overfitting

Example 24 (Forward Selection for Advertising Data)

Define: model score: mean-square-error on the test set for a given model.

1. Fit "Sales $\sim$ Newspaper", "Sales $\sim$ Radio", "Sales $\sim$ TV" and calculate their loss values.
2. Choose the best (lowest loss) single-term model ("Sales $\sim$ TV" in this case), with loss MSE(TV).
3. Fit "Sales $\sim$ TV + Newspaper" and "Sales $\sim$ TV + Radio" and choose the lowest loss score, which is "Sales $\sim$ TV + Radio" with loss being MSE(TV + Radio), which is significantly better.
4. Fit "Sales $\sim$ TV + Radio + Newspaper". Its loss is the same (MSE(TV + Radio) $\approx$ MSE(TV + Radio + Newspaper)), so we favour the existing simpler two-term model (Occam's Razor: other things being equal, choose the simplest model.).

Case Study 3: Advertising: Model Building ("stats" way)

- Start from a "full model" and prune, versus from an "empty model" and add
- We choose the latter, as it is often easier to avoid overfitting

Example 24 (Forward Selection for Advertising Data)

Define: model score: mean-square-error on the test set for a given model.

1. Fit "Sales $\sim$ Newspaper", "Sales $\sim$ Radio", "Sales $\sim$ TV" and calculate their loss values.
2. Choose the best (lowest loss) single-term model ("Sales $\sim$ TV" in this case), with loss MSE(TV).
3. Fit "Sales $\sim$ TV + Newspaper" and "Sales $\sim$ TV + Radio" and choose the lowest loss score, which is "Sales $\sim$ TV + Radio" with loss being MSE(TV + Radio), which is significantly better.
4. Fit "Sales $\sim$ TV + Radio + Newspaper". Its loss is the same (MSE(TV + Radio) $\approx$ MSE(TV + Radio + Newspaper)), so we favour the existing simpler two-term model (Occam's Razor: other things being equal, choose the simplest model.).

So our preferred model is "Sales $\sim$ TV + Radio".

Forward selection in action, with and without the interaction term

Main features only

	feature	test_neg_mean_squared_error	test_r2
0	TV	(-7.324310374422005, -3.936981032219174)	(0.7603440777107349, 0.8390841989031752)
1	Radio	(-4.718440611471557, -1.8510139478354657)	(0.8456097326980663, 0.9322678692463671)
2	Newspaper	(-4.720392592253672, -1.8510521207093062)	(0.8455458626911011, 0.9317779087301497)

MSE(TV) ≈ 5.5; MSE(TV + Radio) ≈ 3.5; MSE(TV + Radio + Newspaper) $\approx 3.5 \approx$ MSE(TV + Radio).
Adding Newspaper does not reduce MSE.

Forward selection in action, with and without the interaction term

Main features only

	feature	test_neg_mean_squared_error	test_r2
0	TV	(-7.324310374422005, -3.936981032219174)	(0.7603440777107349, 0.8390841989031752)
1	Radio	(-4.718440611471557, -1.8510139478354657)	(0.8456097326980663, 0.9322678692463671)
2	Newspaper	(-4.720392592253672, -1.8510521207093062)	(0.8455458626911011, 0.9317779087301497)

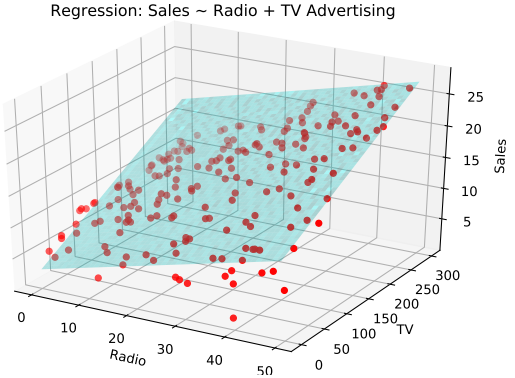
MSE(TV) $\approx$ 5.5; MSE(TV + Radio) $\approx$ 3.5; MSE(TV + Radio + Newspaper) $\approx$ 3.5 $\approx$ MSE(TV + Radio). Adding Newspaper does not reduce MSE.

Main features with TV:Radio interaction term

	feature	test_neg_mean_squared_error	test_r2
0	TV	(-7.324310374422005, -3.936981032219174)	(0.7603440777107349, 0.8390841989031752)
1	TV:Radio	(-3.695048288640372, -1.8479935191656154)	(0.8790957564264389, 0.9377953274242408)
2	Radio	(-3.9297847588258605, -1.7513896129827913)	(0.8714150353235092, 0.9410470781968058)
3	Newspaper	(-3.938746503656725, -1.7715653928145307)	(0.8711218015427205, 0.9403679482294233)

MSE(TV) $\approx$ 5.5; MSE(TV + TV:Radio) $\approx$ 2.8; MSE(TV + TV:Radio + Radio) $\approx$ 2.8 $\approx$ MSE(TV + TV:Radio). Adding Radio and Newspaper does not reduce MSE.

Case Study 3: Advertising: Viewing the Model



Since this two-term model ignores the contribution of the newspaper channel, the Newspaper spend as a contribution to Sales is just another component of the unmodelled (and apparently random) contribution to Sales.

However, the result is a model where every term is highly significant and the model "explains" 90% of the variance of the data, which is high for an observational study. **Why? Can we do better?**

Case Study 3: Advertising: Interactions; Interpretation

- Trying powers greater than 1 of the Radio and TV features did not offer much more.
- However, by adding the TV,Radio interaction so that the model became "Sales $\sim$ TV + TV:Radio" or equivalently "Sales $\sim$ TV * Radio - Radio", the loss decreased significantly, indicating the interaction term is valuable, even more so than the Radio feature.
- All β terms have $t-$statistic significance of approximately 0.001 which is extremely significant.
- $\beta_0 = 6.75$, $\beta_{\text{TV}} = 0.019$, $\beta_{\text{Radio}} = 0.029$ and $\beta_{\text{TV:Radio}} = 0.001$, indicating that there is a favourable relationship between TV and Radio advertising ($\beta_{\text{TV:Radio}} > 0$), and that additional spending on Radio results in more Sales than the same spending on TV ($\beta_{\text{Radio}} > \beta_{\text{TV}}$).
- Spending on Newspaper advertising should be discontinued as its contribution to Sales is insignificant (indistinguishable from random noise).

Case Study 4: Credit balances - overview

Introducing

- the `sklearn` approach to regression (we used `statsmodels` with the Diamonds and Advertising data)
- non-numeric explanatory variables like gender and ethnicity
- more advanced regression modelling, e.g., handling correlated variables

Case Study 4: Credit balances - introduction

	Income	Limit	Rating	Cards	Age	Education	Gender	Student	Married	Ethnicity	Balance
1	14.891	3606	283	2	34	11	Male	No	Yes	Caucasian	333
2	106.025	6645	483	3	82	15	Female	Yes	Yes	Asian	903
3	104.593	7075	514	4	71	11	Male	No	No	Asian	580
4	148.924	9504	681	3	36	11	Female	No	No	Asian	964
5	55.882	4897	357	2	68	16	Male	No	Yes	Caucasian	331

- Note the presence of some categorical features (*Gender*, *Student*, *Married*, *Ethnicity*).
- These can participate in linear regression models to predict a numeric response, but must be coded first.
  - For example, *Gender* can become an indicator (0,1)-valued variable of the form *IsFemale*.
  - *Ethnicity* has 3 levels and is replaced by 3-1=2 indicator variables.

A single categorical feature with n levels becomes n–1 (0,1)-coded "dummy" features.

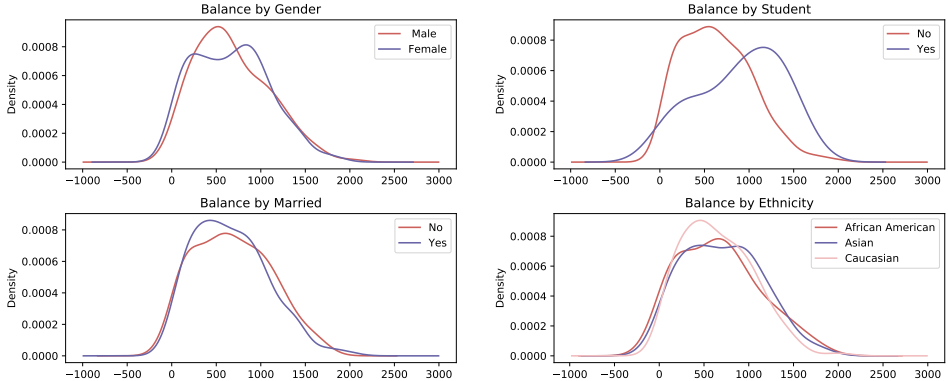
Case Study 4: Credit balances - Removing Data

- the purpose of the analysis is to predict credit balances.
- Basic exploratory data techniques (histograms) soon indicated that there were 2 cohorts
  1. those who do not use their cards and/or clear their balance each month
  2. those who use their cards and have nonzero balances
- Removing data relating to the first cohort meant that the remaining data looked more cohesive and also made linear regression easier
- Take-away: look for inconsistent subsets in the data, if possible: either remove them or develop a separate model for each subset

Case Study 4: Credit balances - Removing correlated feature

- Correlations between predictors are relatively high, but that between “Limit” and “Rating” is 1
- Generally, customers with a high rating are allowed to have high credit limits
- Conversely, customers will not be allowed high credit limits unless they have a high credit rating
- “Limit” was removed from the data used for analysis
- Take-away: remove all but 1 correlated features from a set of such features, because they increase the standard error (hence variance) and make the solver’s job much more difficult (larger condition number)

Case Study 4: Credit balances - Contribution of Categorical Variables



Which of these categorical features has a significant effect on Balance?

Case Study 4: Credit balances - Model building

- Using forward selection as before, the best model was found to be "Balance $\sim$ poly(Income,2) + Rating + Age + Student + Income:Rating"
- Could also use Backward Elimination to prune from a complex model
- For this data, high correlations between features can cause difficulties - we need techniques to handle this

Difficulties caused by correlated features

The Problem : Several features are highly correlated, so the solver has difficulty assigning an importance independently to each.

How it shows up : The condition score is large and several model coefficients take large values with opposite signs. Sometimes the solver gives up.

Solution options :

1. Remove selected features from the model (simple, does not always work and requires care)
2. Use *dimensionality reduction* (linear PCA) to derive an uncorrelated subset of the features with least loss in explanatory power (principal components can be opaque)
3. Use *regularisation*, to "penalise" large model coefficients (solve a related problem with a different loss function)

Regularisation introduction

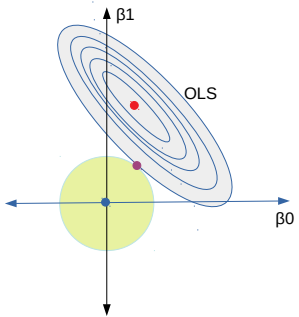
Add *regularisation* constraints to make the model work: $\min_\beta \|\epsilon\|_2^2 + \lambda p(\beta)$

- Options are
  1. *Ridge Regression* where the penalty term takes the form $p(\beta) = \|\beta\|_2$
  2. *Lasso* where the penalty term takes the form $p(\beta) = \|\beta\|_1$
- Regularisation has a metaparameter λ - the challenge is to choose a suitable value
  - if too large: tries less to match the data, increases the bias
  - if too small: tries too hard to match the data so $\beta \to \infty$ and increases the variance

Ridge vs Lasso Regression

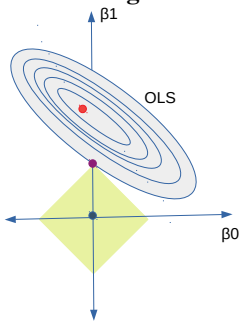
Because lasso regression favours the "corners" in parameter space, it tends to set some parameter values to 0 (essentially dropping the associated features). This has the added benefit of making the model smaller and easier to interpret.

Ridge Regression



Intersection point has $\beta_0 \neq 0$ and $\beta_1 \neq 0$ so both features are needed.

Lasso Regression



Intersection point has $\beta_0 = 0$ so its feature is no longer needed.

Case Study 4: Credit balances - Regularisation - Searching for λ

1. Choose a set of candidate λ values
2. For each candidate λ, use K-fold cross-validation (see Kieran's notes) on data subsets to estimate the prediction error for the regularised fit with that λ
3. Choose the λ for which the expected error is least
4. Now fit all the training data again with this choice of λ

Note that lasso (but not ridge regression) can set particular β_j to 0 (effectively removing them from the model), so it operates more like the *backwards elimination* model building procedure in terms of creating a more frugal model having fewer terms.
Ridge regression downweights certain terms but does not set them to zero. However, it can be more performant, because it keeps some contribution from each feature.

Feature independence in Multivariate Data

Definition 25 (Covariance)

$\sigma_{12} = E[(X_1 - \mu_1)(X_2 - \mu_2)]$. In words, for two features X_1 and X_2, with means μ_1 and μ_2, respectively, σ_{12} is a measure of the the linear dependence between them. If they are independent, we can show that $\sigma_{12} = 0$.

Definition 26 ((Variance-)Covariance Matrix)

When there are n numeric features, there are $n \times n$ pairs of covariances $\sigma_{ij}, i = 1, \ldots, n; j = 1, \ldots, n$. The resulting covariance matrix is symmetric and diagonally dominant. This matrix captures the covariance structure of the set of n features $\{X_i\}$.

- Sometimes it is convenient to work with the correlation matrix, which is a scaled version pf the covariance matrix, with elements $\rho_{ij} = \frac{\sigma_{ij}}{\sigma_i \sigma_j}$, which is scaled so that all the diagonal elements are 1 and the off diagonal elements satisfy $-1 < \rho_{ij} < 1$.
- If two features are highly correlated, adding the second into the model does not increase the explanatory power of the model.
- Therefore, it pays to determine the covariance matrix from the data before building any models.

Multivariate data with correlated measurements

Example 27 (Measles cases, by city, per week from 1948–1985)

This data spans the period before and after the introduction of vaccination for measles (during the mid 1960s). Measles cases are recorded per week in 7 English cities. Although the cities are not adjacent, it is likely that there will be some spatial autocorrelation. Also, by the nature of disease outbreaks, there will also be some temporal autocorrelation per city.

	Date	London	Bristol	Liverpool	Manchester	Newcastle	Birmingham	Sheffield
1	1948-01-17	240	4	51	19	52	84	11
2	1948-01-24	284	3	54	23	34	65	11
3	1948-01-31	340	5	54	31	25	106	4
4	1948-02-07	511	1	89	66	27	142	7
5	1948-02-14	649	3	73	60	47	143	3
6	1948-02-21	766	13	169	87	46	191	6
7	1948-02-28	932	5	212	61	66	208	9

Removing redundant attribues, based on correlation filters

Pearson Correlation

	London	Bristol	Liverpool	Manchester	Newcastle	Birmingham	Sheffield
London	1.000000	0.474016	0.295005	0.519947	0.520185	0.707410	0.539053
Bristol	0.474016	1.000000	0.228214	0.437572	0.374370	0.546398	0.680336
Liverpool	0.295005	0.228214	1.000000	0.431414	0.482269	0.365078	0.329118
Manchester	0.519947	0.437572	0.431414	1.000000	0.554188	0.472575	0.522391
Newcastle	0.520185	0.374370	0.482269	0.554188	1.000000	0.645766	0.535574
Birmingham	0.707410	0.546398	0.365078	0.472575	0.645766	1.000000	0.690961
Sheffield	0.539053	0.680336	0.329118	0.522391	0.535574	0.690961	1.000000

London-Birmingham has correlation greater than 0.7.

Spearman Correlation

	London	Bristol	Liverpool	Manchester	Newcastle	Birmingham	Sheffield
London	1.000000	0.654859	0.399211	0.589346	0.559762	0.764533	0.581148
Bristol	0.654859	1.000000	0.356830	0.598125	0.471088	0.636617	0.613336
Liverpool	0.399211	0.356830	1.000000	0.580160	0.558448	0.383332	0.421292
Manchester	0.589346	0.598125	0.580160	1.000000	0.491076	0.507557	0.577990
Newcastle	0.559762	0.471088	0.558448	0.491076	1.000000	0.591156	0.633679
Birmingham	0.764533	0.636617	0.383332	0.507557	0.591156	1.000000	0.599110
Sheffield	0.581148	0.613336	0.421292	0.577990	0.633679	0.599110	1.000000

London-Birmingham has correlation greater than 0.7.

Kendall Correlation

	London	Bristol	Liverpool	Manchester	Newcastle	Birmingham	Sheffield
London	1.000000	0.471882	0.268666	0.417987	0.402433	0.570474	0.416055
Bristol	0.471882	1.000000	0.243417	0.428664	0.331594	0.460080	0.449481
Liverpool	0.268666	0.243417	1.000000	0.411598	0.400088	0.260798	0.291779
Manchester	0.417987	0.428664	0.411598	1.000000	0.346396	0.354931	0.411831
Newcastle	0.402433	0.331594	0.400088	0.346396	1.000000	0.428067	0.463323
Birmingham	0.570474	0.460080	0.260798	0.354931	0.428067	1.000000	0.432066
Sheffield	0.416055	0.449481	0.291779	0.411831	0.463323	0.432066	1.000000

Observations

- Critical level of correlation $\rho^{(\text{crit})} = 0.7$, so one of London or Birmingham can be dropped.
- The Spearman correlations are particularly high, so more correlation might be present.
- The Kendall correlations are inconclusive.

Working with high-dimensional data

Definition 28 (Curse of Dimensionality)

High dimensions do not just require more computing resources and make interpretation more difficult. They also make it more difficult to capture data that samples very high dimensional spaces efficiently. It can be shown that, as the dimension d increases, most of the volume of a hypercube is near the corners, not near the centre, where data might be easiest to collect. This makes estimating parameters much more difficult.

Working with high-dimensional data

Definition 28 (Curse of Dimensionality)

High dimensions do not just require more computing resources and make interpretation more difficult. They also make it more difficult to capture data that samples very high dimensional spaces efficiently. It can be shown that, as the dimension d increases, most of the volume of a hypercube is near the corners, not near the centre, where data might be easiest to collect. This makes estimating parameters much more difficult.

The proof is based on the fact that, as the dimension d tends to infinity, the *ratio* of the volume of the maximum hypersphere inscribed inside the hypercube of the same dimension, tends to 0. Thus there are good reasons to prefer low dimension approximations to high dimensional space.

Working with high-dimensional data

Definition 28 (Curse of Dimensionality)

High dimensions do not just require more computing resources and make interpretation more difficult. They also make it more difficult to capture data that samples very high dimensional spaces efficiently. It can be shown that, as the dimension d increases, most of the volume of a hypercube is near the corners, not near the centre, where data might be easiest to collect. This makes estimating parameters much more difficult.

The proof is based on the fact that, as the dimension d tends to infinity, the *ratio* of the volume of the maximum hypersphere inscribed inside the hypercube of the same dimension, tends to 0. Thus there are good reasons to prefer low dimension approximations to high dimensional space.

In 2D: imagine the largest circle fitting inside a square; ratio is $\frac{\pi r^2}{4r^2} = \frac{\pi}{4} \approx 0.79$.

Working with high-dimensional data

Definition 28 (Curse of Dimensionality)

High dimensions do not just require more computing resources and make interpretation more difficult. They also make it more difficult to capture data that samples very high dimensional spaces efficiently. It can be shown that, as the dimension d increases, most of the volume of a hypercube is near the corners, not near the centre, where data might be easiest to collect. This makes estimating parameters much more difficult.

The proof is based on the fact that, as the dimension d tends to infinity, the *ratio* of the volume of the maximum hypersphere inscribed inside the hypercube of the same dimension, tends to 0. Thus there are good reasons to prefer low dimension approximations to high dimensional space.

In 2D: imagine the largest circle fitting inside a square; ratio is $\frac{\pi r^2}{4r^2} = \frac{\pi}{4} \approx 0.79$.

In 3D: imagine the largest sphere fitting inside a cube; ratio is $\frac{(4/3)\pi r^3}{8r^3} = \frac{\pi}{6} \approx 0.52$.

Working with high-dimensional data

Definition 28 (Curse of Dimensionality)

High dimensions do not just require more computing resources and make interpretation more difficult. They also make it more difficult to capture data that samples very high dimensional spaces efficiently. It can be shown that, as the dimension d increases, most of the volume of a hypercube is near the corners, not near the centre, where data might be easiest to collect. This makes estimating parameters much more difficult.

The proof is based on the fact that, as the dimension d tends to infinity, the *ratio* of the volume of the maximum hypersphere inscribed inside the hypercube of the same dimension, tends to 0. Thus there are good reasons to prefer low dimension approximations to high dimensional space.

In 2D: imagine the largest circle fitting inside a square; ratio is $\frac{\pi r^2}{4r^2} = \frac{\pi}{4} \approx 0.79$.

In 3D: imagine the largest sphere fitting inside a cube; ratio is $\frac{(4/3)\pi r^3}{8r^3} = \frac{\pi}{6} \approx 0.52$.

More generally

$$\frac{V_{\text{hypersphere}}}{V_{\text{hypercube}}} = \frac{\pi^{d/2}}{d2^{d-1}\Gamma(d/2)} \to 0 \text{ when } d \to \infty$$

Working with high-dimensional data

Definition 28 (Curse of Dimensionality)

High dimensions do not just require more computing resources and make interpretation more difficult. They also make it more difficult to capture data that samples very high dimensional spaces efficiently. It can be shown that, as the dimension d increases, most of the volume of a hypercube is near the corners, not near the centre, where data might be easiest to collect. This makes estimating parameters much more difficult.

The proof is based on the fact that, as the dimension d tends to infinity, the *ratio* of the volume of the maximum hypersphere inscribed inside the hypercube of the same dimension, tends to 0. Thus there are good reasons to prefer low dimension approximations to high dimensional space.

In 2D: imagine the largest circle fitting inside a square; ratio is $\frac{\pi r^2}{4r^2} = \frac{\pi}{4} \approx 0.79$.

In 3D: imagine the largest sphere fitting inside a cube; ratio is $\frac{(4/3)\pi r^3}{8r^3} = \frac{\pi}{6} \approx 0.52$.
More generally

$$\frac{V_{\text{hypersphere}}}{V_{\text{hypercube}}} = \frac{\pi^{d/2}}{d2^{d-1}\Gamma(d/2)} \to 0 \text{ when } d \to \infty$$

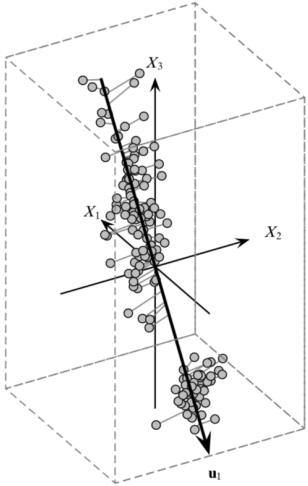
Impact: Harder to collect data that samples high-dimensional space, so harder to estimate such models.

Feature reduction

- Sometimes it is possible to use intuition to reduce the dimension, by omitting selected features.
- Another possibility is to look for groups of correlated features (c.f., *mediation*), such as the London and Birmingham measles cases above, and just choose 1 of these.
- More generally, there are techniques that search for a subspace with specified dimension d' of the features that captures most of the variance of the full set of features having dimension d, where $d' < d$ (often $d' \ll d$).
- The best known of these techniques is *Principal Components Analysis* (PCA).

PCA visualisation

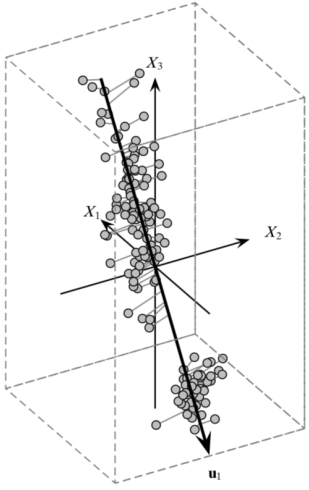
Mapping to 1 Principal Component



Mapping correlated X_1, X_2, X_3 to uncorrelated u_1

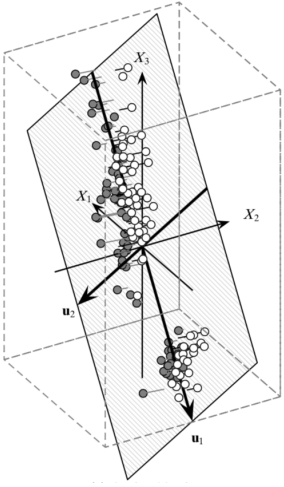
PCA visualisation

Mapping to 1 Principal Component



Mapping correlated X_1, X_2, X_3 to uncorrelated u_1

Mapping to 2 Principal Components

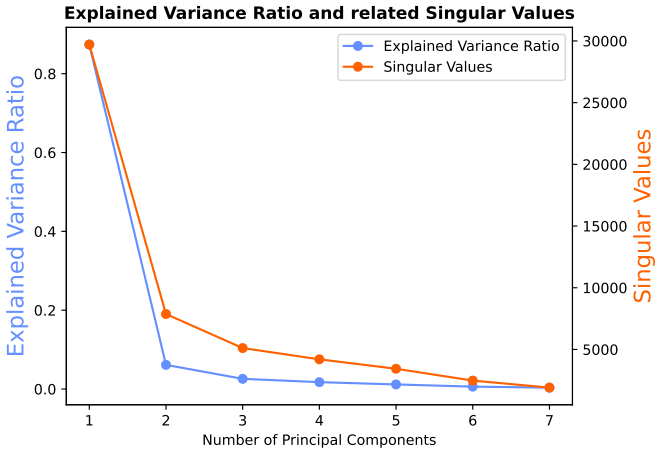


Mapping correlated X_1, X_2, X_3 to uncorrelated u_1, u_2

PCA interpretation

- Although the data has dimension $d = 3$, it is possible to find the line (on the left; $d = 1$) and plane (on the right, $d = 2$) which retain most of the variance of the data after it has been projected onto this lower dimensional subspace.
- First compute the transformations needed to align the training data with the selected subspace.
- Train the model using the transformed training data and the transformed features (principal components of the original features)
- Can then project other data, e.g., test data, onto the subspace that was derived with the original, training data, and use the model to perform predictions in the transformed space.
- Apply the inverse projection to the data, restoring it to its original orientation. However, because of the use of projections, it is not the same as the original data - the round-trip is "lossy".
- However, it is helpful to interpret the results in terms of the original features.

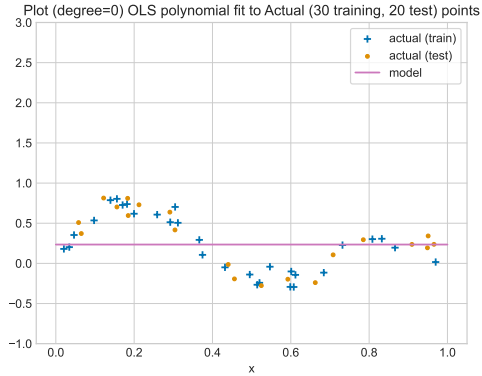
PCA example



The plot shows that the first 3 singular values (associated with principal components u_1, u_2, u_3) capture the bulk of the variance in the training set. Therefore, three features, which are transformations of the other 7, are sufficient. You could interpret those features as representing the measles outbreaks in three archetypal English cities…

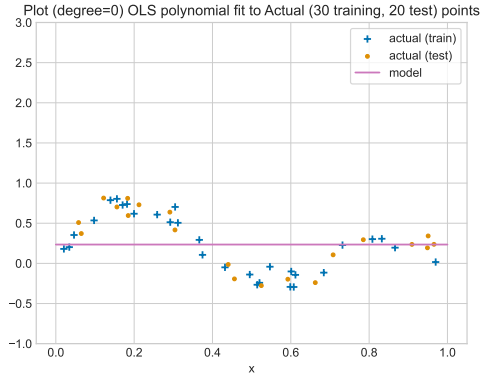
This youtube video describes PCA concepts well.

Returning to the problematic example

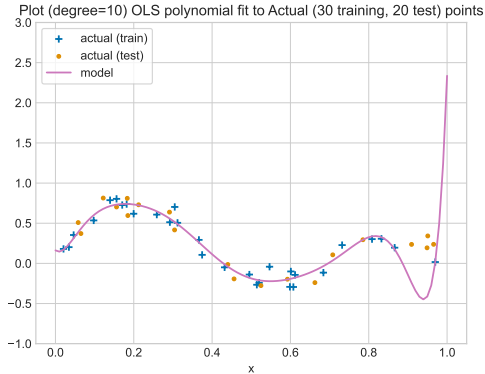


Degree 0 (constant) fit: high bias, low variance

Returning to the problematic example

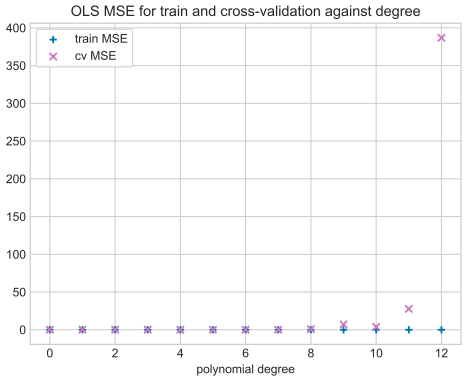


Degree 0 (constant) fit: high bias, low variance



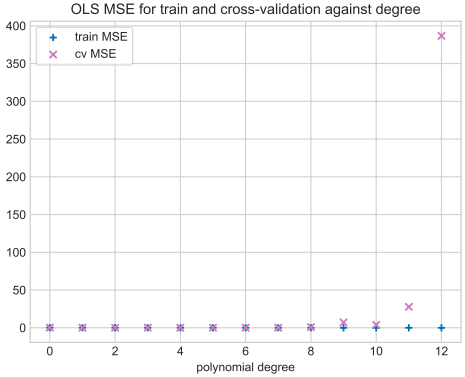
Degree 10 (up to x^{10}) fit: low bias, high variance

Diagnosis - OLS

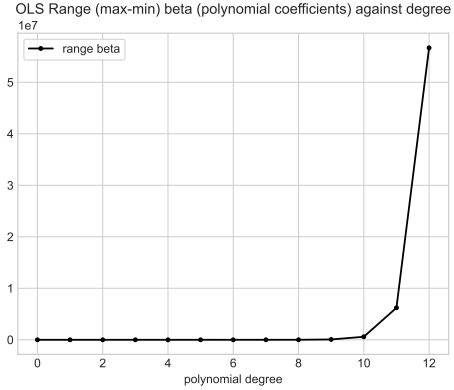


Train MSE decreases with degree, Test MSE decreases, then increases

Diagnosis - OLS

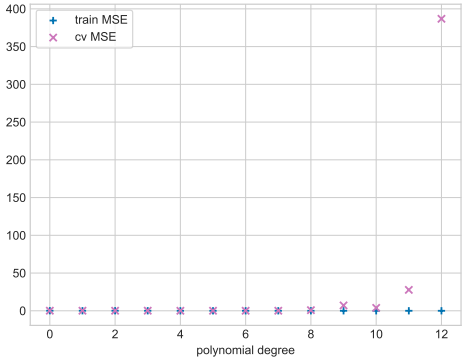


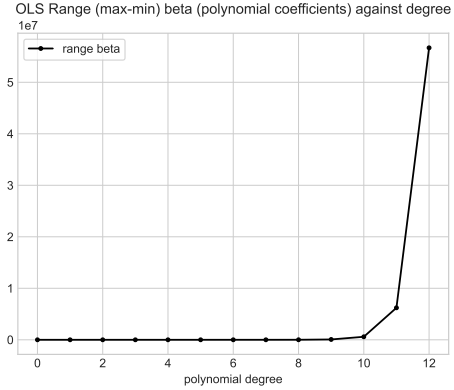
Train MSE decreases with degree, Test MSE decreases, then increases



Polynomial coefficient range (max-min) increases dramatically with degree due to overfitting.

Diagnosis - OLS



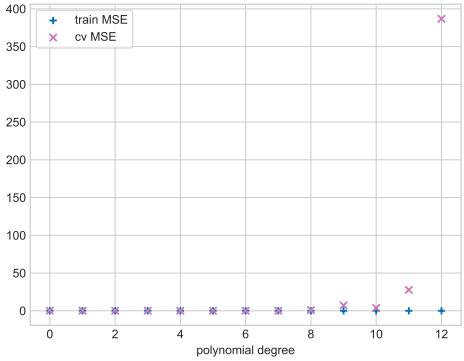


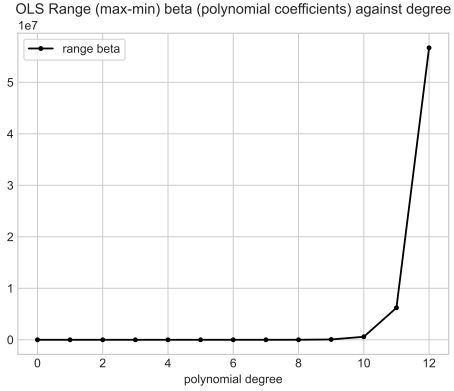
Train MSE decreases with degree, Test MSE decreases, then increases

Polynomial coefficient range (max-min) increases dramatically with degree due to overfitting.

Is there any way we can use high-degree polynomials?

Diagnosis - OLS





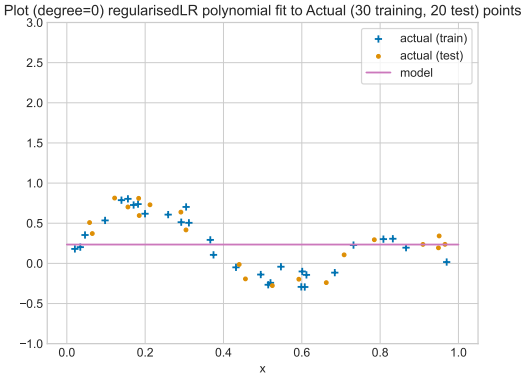
Train MSE decreases with degree, Test MSE decreases, then increases

Polynomial coefficient range (max-min) increases dramatically with degree due to overfitting.

Is there any way we can use high-degree polynomials?

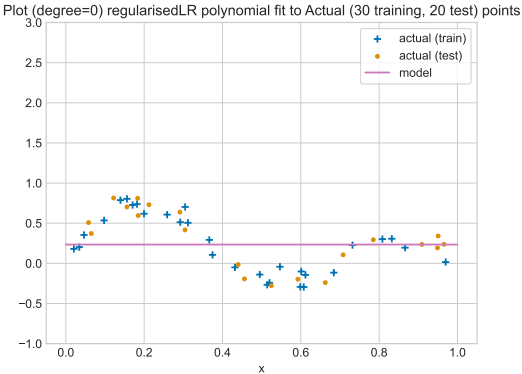
Yes, if we add regularisation…

Same data, same features, with regularisation this time

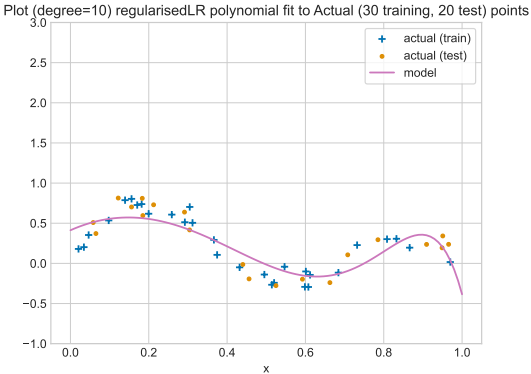


Degree 0 (constant) fit, $\lambda \approx 0$: no change

Same data, same features, with regularisation this time

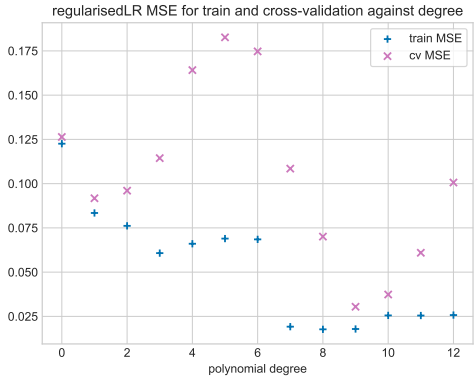


Degree 0 (constant) fit, $\lambda \approx 0$: no change



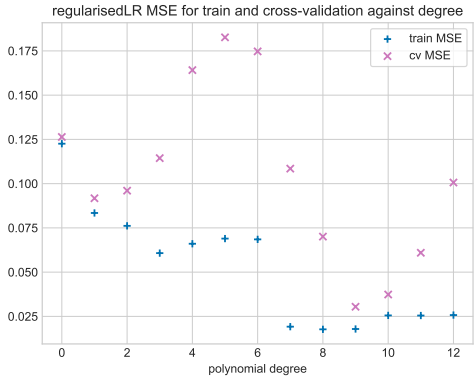
Degree 10 (up to x^{10}) fit: stabilised polynomial

Diagnosis - Regularised Linear Regression

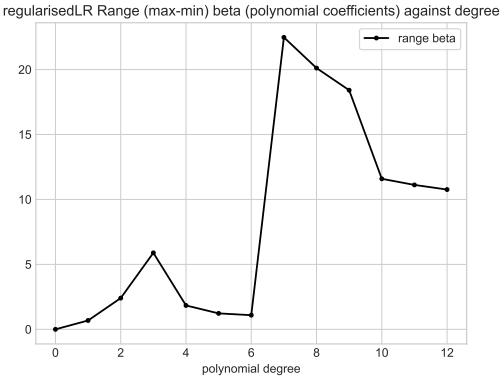


MSE behaviour is affected by choice of λ, but degree 8 or 9 looks good

Diagnosis - Regularised Linear Regression

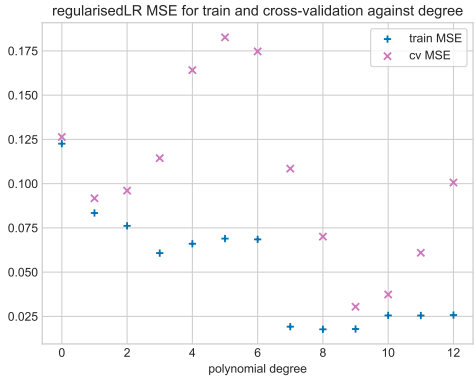


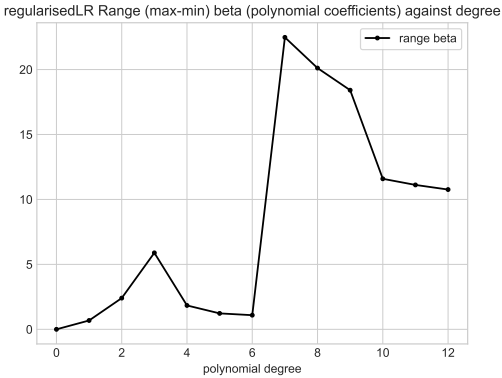
MSE behaviour is affected by choice of λ, but degree 8 or 9 looks good



Polynomial coefficient range (max-min) is controlled - no evidence of overfitting.

Diagnosis - Regularised Linear Regression





MSE behaviour is affected by choice of λ, but degree 8 or 9 looks good

Polynomial coefficient range (max-min) is controlled - no evidence of overfitting.

So regularisation can control overfitting and/or high correlation between features

Review and summary

- Linear regression is one of the classic machine learning techniques.
- Compared to other techniques, statistics have more to offer, but ML objective (minimise prediction error on test set) is still as important!
- It has two phases, of which the first (learning from the training set) is generally the most challenging.
- It has many variants, so is quite flexible, but flexibility can be abused!
- Careful validation and model building is essential for success - it is an extension of the exploratory work done earlier in the process.
- In machine learning, prediction error is the main focus, but you need to be aware of other considerations such as
  1. model parsimony (keep model as small/simple as possible!): faster at both training and evaluation time
  2. the bias-variance dilemma: avoid overfitting and underfitting - remember, your model needs to generalise well from the training to the test set
  3. model interpretability: some models are easier to understand because the terms in the model represent concepts from the domain the data is from

Some Additional Resources

- Book: Introduction to Statistical Learning with R (2013) by James, Gareth and Witten, Daniela and Hastie, Trevor and Tibshirani, Robert.

  I strongly recommend that you read Chapter 3 of the book, as it is very well written and available online for free.
- Kaggle notebooks relating to the datasets addressed this week. There are many, but searching Kaggle should provide nice examples of data mining in action.
- I uploaded a background report on linear regression that is available for download from here.