**Statistical Thinking**

The formulation of a good research question is essential ***before*** beginning any research and associated data sourcing. It is a systematic process which needs to be carried out meticulously as any results could guide clinical decision making or business actions toward the larger customer base.

We can categorise research questions as:

**1.)** **Comparative** - comparing differences between two or more groups

*"Is there a difference in 0-3 year survival between elec-only customers who joined the Rewards Club at sign-up and those who never joined it?"*

Dependent variable: customer survival

Groups: elec-only customers that joined the Rewards Club at sign-up, elec-only customers who never joined the Rewards Club.

*"Is there a difference in first cycle 12-month contract renewal rates between Dublin-based Rewards Club customers that received complimentary theatre tickets along with the standard renewal call and those that received the renewal call only?"*

Dependent variable: 12-month contract renewal rate

Groups: First cycle 12-month dual-fuel contract customers in Rewards based in Dublin with complimentary tickets, first cycle 12-month dual-fuel contract customers in Rewards based in Dublin without complimentary tickets.

**2.)** **Causal** - investigating if a variable causes one or more outcomes; cause-and-effect inference; relationship research

*"What is the relationship between take-on channel and new customer survival to Q4 of tenure?"*

Dependent variable: customer survival

Independent variable: time

Strata: take-on channels

Groups: new customers from each take-on channel

**3.)** **Descriptive -** describing something

*"What is the average customer lifetime value for elec-only customers that signed up through the Bonkers channel?"*

Variable: customer lifetime value

Group: elec-only customers via Bonkers

*"How much energy, on average, do residential gas BGE customers consume annually?"*

Variable: annual gas consumption (kWh)

Group: all BGE residential gas customers

*"What factors increase the risk of churn in BGE twin-fuel customers?"*

Dependent variable: risk of churn (e.g. a hazard ratio or odds ratio)

Independent variable(s): this is what we want to know

Group: all BGE twin-fuel customers

***Once we have a well formulated research question, an appropriate statistical analysis can be conducted***.

Examples of poor research questions:

*"Having a free boiler service stops churn."* - This is too broad because tenure has a large impact on churn. Also, *"stops"* is far too strong an effect to hope to uncover as some customers who receive boiler services will no doubt churn anyway.

*"GAA engagement in 2017 will reduce churn by 3,000 SAs."* - This is so broad that is impossible to measure and ignores the time-to-event nature of BGE customer survival data. Does it refer to Q4 renewals, Q8 renewals, churn at any point in tenure, contract renewals, or all of them? Is there a control? If not, no controls no conclusions.

Better versions of these research questions, that could be tested empirically, would be:

*"A free boiler service in Q4 of tenure reduces the risk of churn in BGE residential gas customers over a two-year study period."*

*"Over a 1-year study period, is there a difference in survival between rural Rewards Club customers who engaged with GAA in 2017 and those that did not, controlling for all other factors?"*

**Sampling methods**

***A sample is a subset of data that should reasonably mimic the population from which it was drawn***. In most cases, we cannot obtain data on the entire population of interest (e.g. smokers under 18 in the entire world).

In BGE, however, we can potentially calculate descriptive statistics for the entire population of interest (e.g. all BGE residential electricity customers). Remember that this population itself only makes up a sample of the wider population of electricity customers in the country of Ireland.

Obtaining a representative sample is an essential step in statistical research. It must be objective, and bias cannot be introduced. Common types of bias are:

* selection bias (convenience sampling)
* confirmation bias (sampler wants to prove they are correct)
* response-driven bias (response, non-response)

So how do we ensure an objective unbiased sample is taken? ***Randomisation*** and ***replication***. With these two Rs, you will not go far wrong. Randomisation reduces bias as each subject has an equal probability of being randomised into treatment/control groups. Replication increases reliability.

Another very important concept is that of ***initial conditions***. When conducting comparative research, for example, the only systematic difference between groups should be the treatment(s) of interest. Randomisation ensures this.

**Simple Random Sampling**

An example would be randomly assigning 100 Dublin-based Rewards Club customers into a treatment group and 100 Dublin-based Rewards Club customers into a control group to trial a new loyalty offering and its effect on churn (i.e. customer survival).

**Stratified Sampling**

An example would be splitting new customer data by take-on channel and then performing simple random sampling on each split to obtain 500 customers from each channel. We could then carry out descriptive research on median lifetime by channel with *representative* samples.

**Cluster sampling**

An example would be using each county in Ireland as a cluster and within each cluster randomly sampling customers.

Sample size is very important and can impact *p*-values (i.e. statistical significance) in statistical tests and models. Power analysis determines an adequate sample size and should always be carried out if sample size is a concern due to limited data resources.

**Correlation and Linear Regression**

Correlation

Correlation is used to measure ***the degree of association between two variables***. However, note that correlation does not imply causation. For causality to be inferred, a controlled experiment is required.

There are two measures of correlation.

Pearson’s and Spearman’s Rank Correlation Coefficients

Both measure the strength of association between two variables. Pearson’s is ***parametric*** whereas Spearman’s Rank is ***non-parametric***.

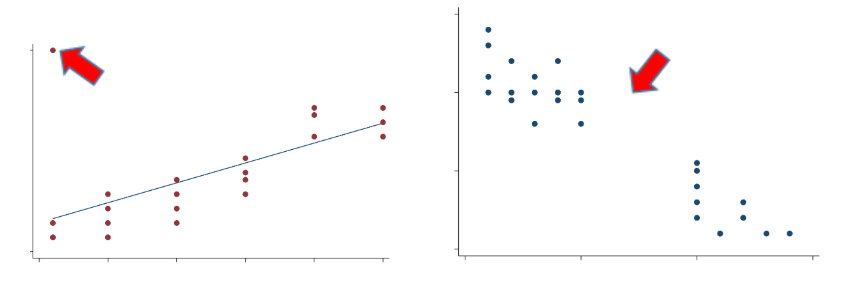
Both Pearson’s and Spearman’s give values between -1 (strong negative association) and +1 (strong positive association). Thresholds for weak, moderate and strong associations are somewhat arbitrary.

There are a few important differences you need to be aware of before you decide which, if any, is best for your data:

1. Pearson’s uses the actual observed values to calculate the correlation, but Spearman’s ranks the values and then calculates the coefficient using the ranks;
2. Pearson’s measures ***the strength of a linear association*** between two variables, whereas Spearman’s is not quite as restrictive and measures if there is an association ***assuming only a monotonic relationship***.
3. Pearson’s requires that both variables are continuous, whereas Spearman’s can be used for continuous and ordinal variables.
4. Pearson’s requires that both variables are approximately normally distributed in the population, whereas Spearman’s makes no such distributional assumptions. Both require that observations are a random sample from the population of interest.

A word of warning

Remember to always examine your data in a *scatterplot* before calculating a correlation. Not only will it allow you to assess the assumptions, but it also helps you spot outliers and discontinuities in the data.



It does not matter which variable goes on the x- or y-axis when analysing correlation but it does matter when using linear regression (dependent vs. independent variable(s)).

Linear regression

Correlation tells you if you have a strong, moderate or weak association between two variables, but it doesn’t describe or quantify that relationship. If you want to quantify the impact of a change in one variable on another, you can use linear regression.

Previously, we looked at whether height was a predictor of lung function (the outcome or dependent variable). It’s natural to think of height as influencing lung function. It’s not always obvious when variables are measured at the same time-point which one is the outcome, and for these situations, it is something you will need to decide. Once you’re happy with your research question, you can proceed.

***Linear regression is simply a way of fitting an optimal straight line to your data***; it allows you to evaluate relationships and make predictions.

One way to fit an optimal straight line to the data is to minimise the sum of squared residuals, also referred to as the ***residual sum of squares*** (RSS).

– the value of the outcome variable

– the value of the predictor variable

– the intercept; the value of when the value of = 0

– the coefficient of the slope; the linear effect of the predictor on the outcome

– the error (i.e. residual) as the model is not expected to predict every *y* value exactly

Software such as R can be used to find *α* and *β*, such that ***the line minimises the squared distances between each observed value and the predicted regression line***. The units of the sum of squared errors (SSE) can be hard to interpret so the root mean squared error (RMSE) is used and is normalised to the units of the dependent variable.

Recall:

* *α* is the intercept of the straight line which is the value of the outcome where the line cuts the *y*-axis i.e. when the predictor equals 0;
* *β* is the slope, or the gradient, of the line and it quantifies the linear relationship between your two variables. ***It tells you how much, on average, your outcome variable increases for a one unit increase in your predictor***.

For the model to be valid, there are 3 key assumptions that need to be satisfied:

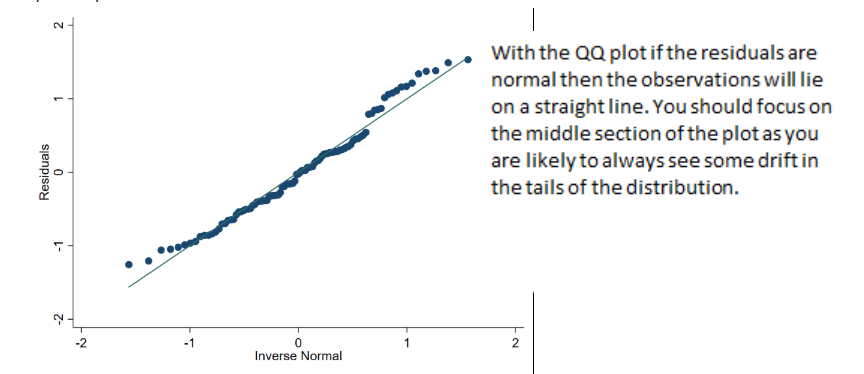
1. Linearity between outcome (*y*) and predictor (*x*) variables
2. The outcome variable is normally distributed across predictor values
3. The variance of outcome is the constant across predicted values

***If these assumptions hold, then your residuals are normally distributed with a mean of 0 and constant variance (i.e. homoscedasticity) across the predicted values.***

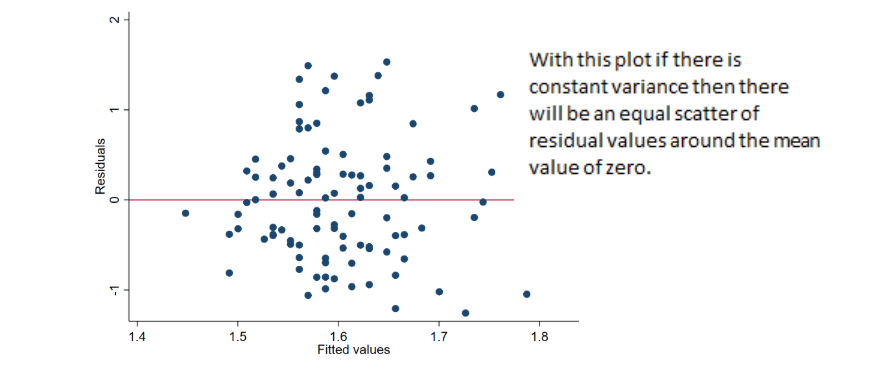
Residuals ~ Normal (0, *σ*2)

There are two useful plots that can help you assess model assumptions.

1. QQ plot (for assessing normally distributed residuals)



**2. Scatterplot of the residuals by the fitted regression values (for constant variance)**



Multivariate linear regression (multiple linear regression)

The simple linear regression model can be extended to include multiple predictor variables:

*y* = *α* + *β*1*x*1 + *β*2*x*2 + ... + *β*p*x*p + *e*

Remember that this slightly alters your interpretation of the *α* and *β* coefficients.

* *β*1 is now the average change in *y* for every one unit increase in *x*1 keeping *x*2 held constant;
* *β*2 is the average change in *y* for every one unit increase in *x*2 keeping *x*1 held constant;
* And so on for *p* predictors.

You can investigate ***collinearity*** amongst the predictor variables, another important consideration in linear regression, by examining the correlations between pairs of independent variables or through ***variance inflation factors*** (VIFs).

Correlations are limited to only examining pairwise relationships. Correlations don’t allow you to identify dependence amongst three or more predictors. The variance inflation factor can help you identify the collinearity in this situation. ***The VIF quantifies how much the variances of coefficients are inflated by***, ***so each coefficient will have a VIF***. VIFs equal to 1 indicate no collinearity amongst the predictors in the model, values above 4 would prompt further investigation, and anything above 10 would indicate serious issues of collinearity. There are some situations in which you don’t need to worry about collinearity, but it is always important to check.

Good practice steps to develop a multivariate linear regression model are:

1.) ***Inspect your variables using summary statistics, histograms for continuous variables and frequency tables for categorical variables. Normal distributions and non-zero variance are important.***

This helps you identify and quantify the amount of missing information. It also helps you spot any strange or outlying observations. If you do spot any strange values, you will need to decide what to do with them before you move on.

2.**)** ***Examine the relationship between your candidate predictor variables using cross tabulations for categorical variables and pairwise correlations and scatterplot matrices for continuous variables.***

This helps you identify potential associations across your candidate predictor variables that could be problematic if included together in your multivariate model.

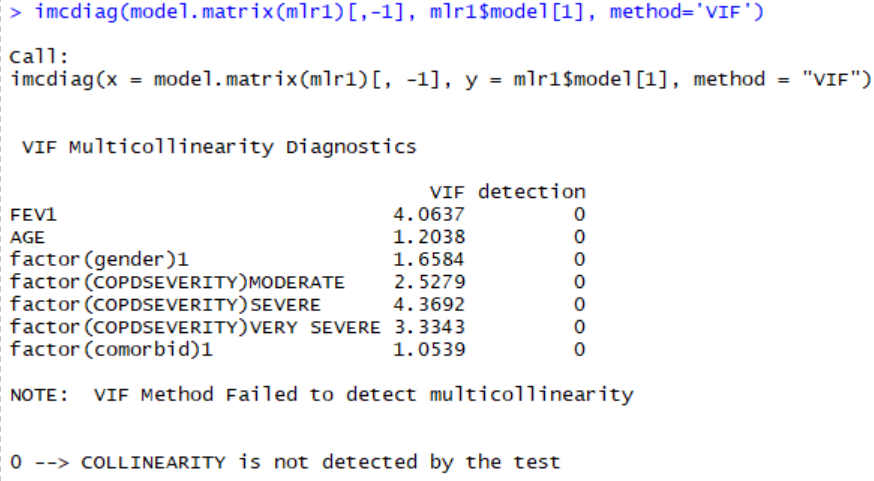
3.**)** ***Fit a simple linear regression model between the outcome and each of your candidate predictor variables.***

This allows you to assess the relationship between each of your candidate predictor variables and the outcome variable. Again, this helps to spot errors in either the data or the coding, and it also allows you to anticipate what you might expect to happen when you fit the multivariate model. Any lack of change in most of the coefficients between the simple models (save the coefficients for comparison) and the multivariate model should reassure you that collinearity is unlikely to be a problem in the multivariate model. This can also be explored by examining the VIFs.

Once you’ve taken these steps, you’re in a much better position to start thinking about building a multivariate regression model for production or reporting. Just like simple linear regression with one covariate, in multivariate regression the best coefficients are selected to minimise the SSE.

To summarise**, *it’s important that you get to know your data well before you start modelling***.

Below is an example analysis of VIFs.



This suggests potential problems with FEV1 and COPDSEVERITY, and in fact it does not seem unreasonable that FEV1 (a measure of lung capacity) and COPDSEVERITY rating might be associated. A decision needs to be made about which covariate to keep in the model. It is important to check univariate coefficients and how they change in the multivariate model.

***Regression models don’t always have to assume that predictor effects are additive. Instead predictor effects can depend on each other (i.e. there can be interactions between predictors).***

As an example, we can look at the effect of an interaction between two binary predictor variables, atrial fibrillation and diabetes on walking distance:

MWT1best = *α* + *β*1∗Diabetic + *β*2∗AtrialFib + *β*3∗Diabetic∗AtrialFib

where:

Diabetic = 0 if diabetes not present, 1 if present; (it is always good to use 0 for the absence case)

AtrialFib = 0 if atrial fibrillation not present, 1 if present;

Diabetic\*AtrialFib = 0 if diabetes or atrial fibrillation are not present, and 1 if diabetes and atrial fibrillation are present.

To fit this model in R you first need to create the Diabetic\*AtrialFib variable.

Note: R will not be able to create this new variable if the Diabetes and AtrialFib variables are saved as factors. They therefore need to remain/be changed to integers. You can do this using the as.integer() function:

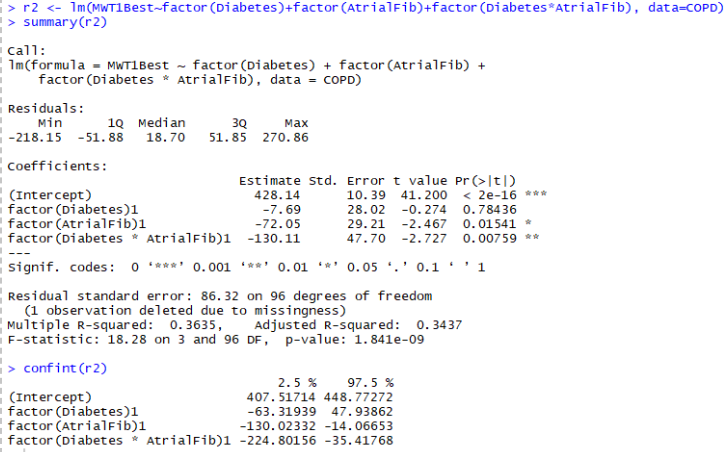
You can create the new variable using the R code:

DAF <- COPD$Diabetes \* COPD$AtrialFib

You can then code for the regression, using the command:

r1 <- lm(MWT1Best ~ factor(Diabetes) + factor(AtrialFib) + factor(DAF), data = COPD)

You can view the regression using the function summary(r1) and the confidence intervals using the command confint(r1).



which equates to:

MWT1best = 428.1 − 7.7∗Diabetic −72.0∗AtrialFib − 130.1∗(Diabetic∗AtrialFib)

This can be interpreted as follows:

A person with diabetes and no atrial fibrillation has an estimated average walking distance of 420.4 metres.

MWT1best= 428.1 − 7.7∗Diabetic

= 428.1 − 7.7

= 420.4

A person with atrial fibrillation and no diabetes has an estimated average walking distance of 356.1 metres.

MWT1best = 428.1 − 72.0∗AtrialFib

= 428.1 − 72.0

= 356.1

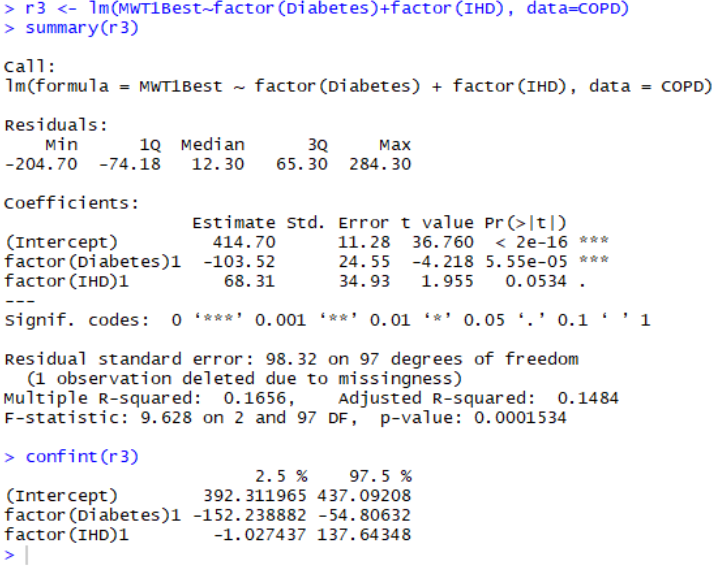
A person with both diabetes and atrial fibrillation has an estimated average walking distance of 218.3 metres. ***Include all coefficients in the equation!***

MWT1best =

428.1 − 7.7∗Diabetic − 72.0∗AtrialFib − 130.1∗(Diabetic∗AtrialFib)

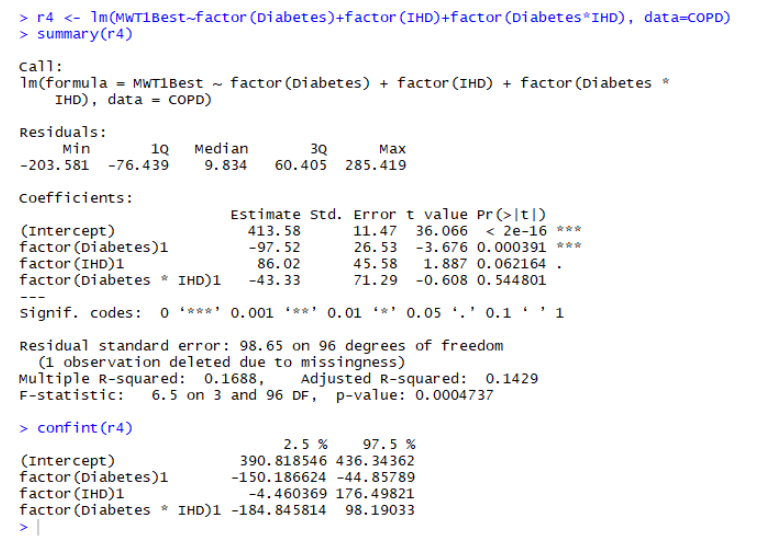
= 428.1 − 7.7 − 72.0 − 130.1 = 218.3

Now look at the model with the interaction effect of having IHD and diabetes. First, fit the model without the interaction effect: MWT1best = *α* + *β*1∗Diabetic + *β*2∗IHD



Now, you might suspect that the predictive effects of diabetes and IHD depend on each other. You can investigate this by including the interaction term:

MWT1best = *α* + *β*1∗Diabetic + *β*2∗IHD + *β*3∗Diabetic ∗ IHD



The effect of the interaction term in this model has been far less dramatic on the regression coefficients. This suggests the lack of an association between IHD and diabetes. In fact, there is no evidence of an interaction effect between IHD and diabetes.

Finally, a measure of model quality is *R*2. This represents the variance in *y* explained by the variance in *x*. It compares the fitted model to a baseline model where all values of the covariates equal zero (i.e. the line is horizontal through the plotted data at the mean of *y*). *R*2 is unitless and reveals the value that a model adds in reducing the squared errors. Adjusted *R*2 is better to use when the model is multivariate but be careful not to add too many covariates as this can lead to overfitting.

**Survival Analysis**

Our final chapter concerns models for the analysis of data which have three main characteristics: (1) the dependent variable or response is the waiting time until the occurrence of a well-defined event, (2) observations are censored, in the sense that for some units the event of interest has not occurred at the time the data are analysed, and (3) there are predictors or explanatory variables whose effect on the waiting time we wish to assess or control. We start with some basic definitions.

**KEY CONCEPTS**

* Define the branch of statistics known as survival analysis
* Explain when it is valid to use survival analysis
* Run the Kaplan-Meier plot and log-rank test in R and interpret the results

Survival analysis can be applied to studies that run over many months or years. The anatomy of a longitudinal study is comprised of a planning phase, a data collection (or enrolment) phase with a start and end date, a follow-up date (end of study; usually sometime after follow-up) and a data analysis phase.

We are mainly interested to know when a particular outcome has occurred over the study period. The **“if”** and the **“when”** rather than just the “if” as with logistic regression and classification techniques.

Examples of outcomes survival analysis deals with:

* Time to onset of speech from birth
* Time to cancer relapse
* Time to hospital discharge after a kidney transplant
* Time to customer churn
* Time to mechanical failure of an instrument

**Life tables: the basics**

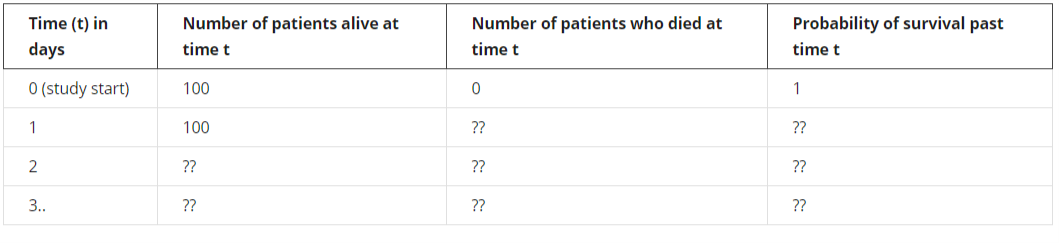
Life tables are used to measure the probability of death (or the event occurring) at a given age (or time) and the life expectancy at varying ages/times. Actuarial science and of course life insurance companies need to know this in detail, but we in public health do too. There are two different kinds of life table:

* Cohort or generational life tables
* Current or period life tables

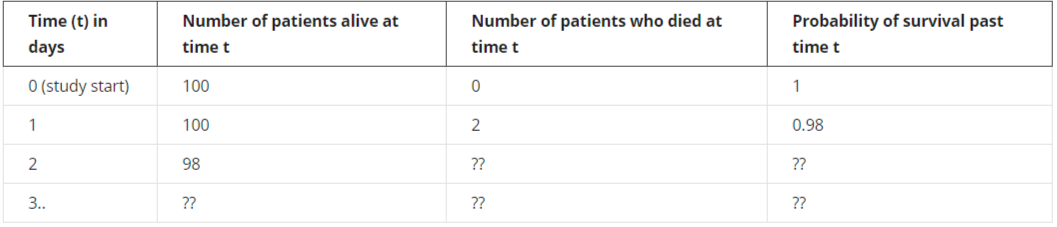
Cohort life tables take an actual set of people born at the same time, usually in the same year or even on the same day of the same year and follow them up for their whole lives. Several countries, including Norway, Denmark and the US, have these “birth cohorts” such as the Millennium Cohort Study in the UK that follows up people born in 2000. The mortality experience of such a cohort teaches us a lot and is great for history, but it’s unlikely to be completely relevant to people born at other time points.

Period life tables take a hypothetical cohort of people born at the same time and uses the assumption that they are subject to the age-specific mortality rates of a region or country. These rates are often calculated using census data as the base population and actual age-specific death rates during the census year (and typically also one year either side).

How are life tables constructed? In a common type of epidemiological study called **a cohort study**, a set or cohort of patients are enrolled at time zero and then followed up to see who gets the outcome of interest, such as death, and when they get it. The latter will often be measured in days since the study start, but not necessarily. In theory you could measure it in milliseconds, but that’s silly unless you’re looking at something like biochemical reactions. At time zero, a table of the numbers of people with and without the outcome at each time point will look like this. Let’s suppose that we start off with 100 patients.

Everybody makes it past time zero, so the probability of surviving at least to time *t* = 0 is 1, or 100%. This probability is technically known as the **survival function**, one of two core concepts in survival analysis. The survival function uses the cumulative survival probability.

Let’s now say that two people die the day after they are enrolled. The life table then looks like this:

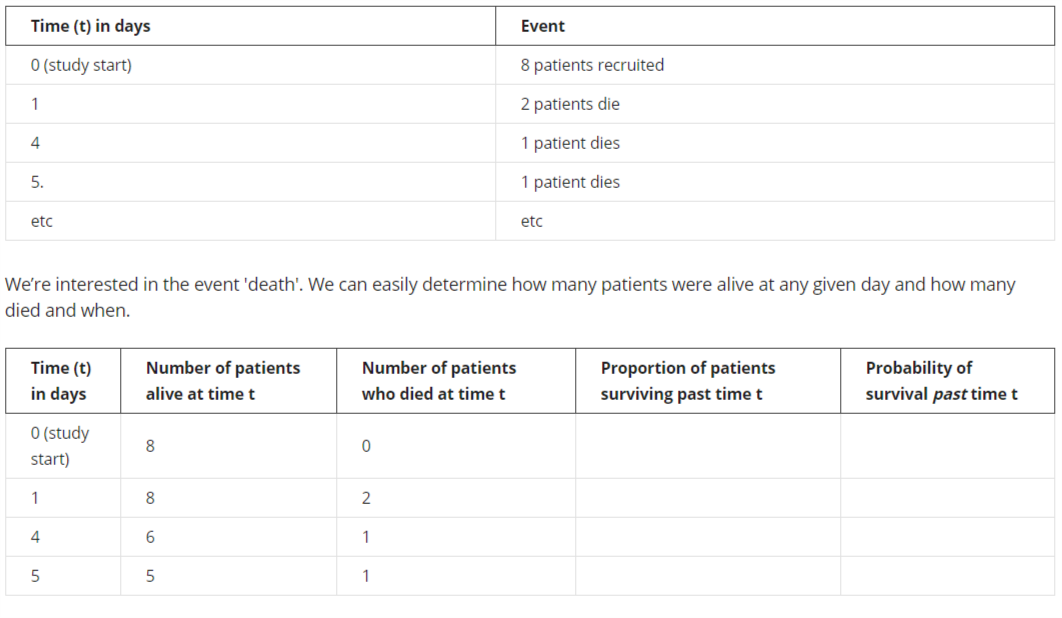
****

The calculations continue in that way. However, this assumes that everybody enters the study at the same time, *t* = 0, and no one leaves it except by death. It ignores the more realistic case when people drop out or are “lost to follow-up”. The technical term for this is that these people are **censored**. Censoring is an important concept in survival analysis. There are different forms, but the type due to people dropping out – or when people are still alive at the study end – is the most common (this is called right-censoring). The Kaplan-Meier table and associated plot is the simplest (but not the only) way of estimating the survival time when you have drop-outs.

**How to calculate a Kaplan-Meier table and plot by hand**

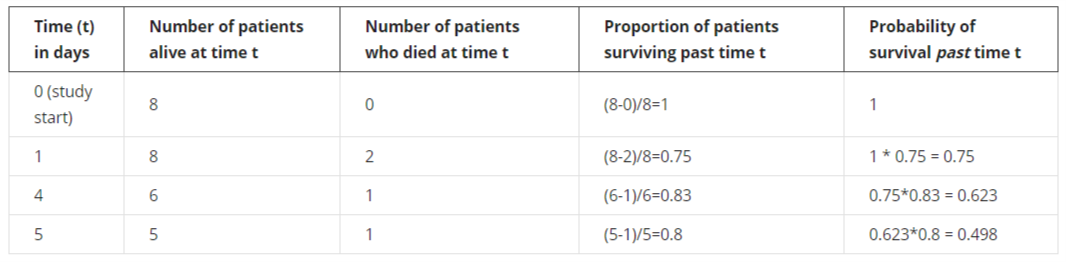
The plot of the survival function versus time is called the survival curve. The Kaplan-Meier method can be used to estimate this curve from the observed survival times without the assumption of an underlying probability distribution (it is, therefore, *non-parametric*). Other kinds of survival analysis such as Weibull do require estimating the underlying distribution for the survival times. One reason why the KM method is so popular is that it doesn’t make any such assumptions. **Whenever you make assumptions in statistics, you must test whether they are valid**.

To better understand the Kaplan-Meier method we’ll now use it to draw a survival curve. Let’s suppose we are monitoring patients after a treatment. After 5 days of follow-up we have the following information (example adapted from [https://www.ncbi.nlm.nih.gov/pmc/articles/PMC1065034](https://www.ncbi.nlm.nih.gov/pmc/articles/PMC1065034/)).

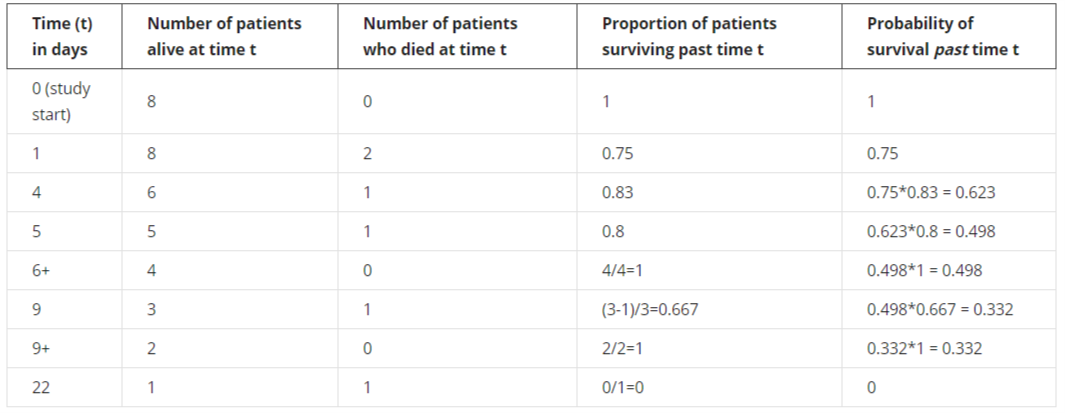
****

But how do we compute the probability of survival past time *t*? Start by computing the proportion of patients that survive day, i.e. of those alive at the beginning of day *t*, what proportion make it to the next day alive? On day 0, the day the study begins, there are no deaths. Everybody survives. Hence the proportion surviving is 1. On the following day, 2 out of 8 patients don't make it; in other words, 75% survive the day. The wording is thus “***the probability of surviving past 1 day is 75%***”.

Now we know the proportions, but what are the probabilities? With no deaths on day 0, the probability of surviving is 1. Computing the next probability is a bit trickier. The basic idea underlying Kaplan-Meier tables comes into play here: the probability of surviving past day *t* is simply the probability of surviving past day *t*-1 times the proportion of patients that survive on day *t*. Let's see it together:

****

Adding more of the data, we now have some drop-outs, i.e. patients whose outcome we don't know exactly. These patients are censored and should be treated differently from patients that die. When a patient is censored at time *t*, we know the patient was alive at time *t*, but we don't know whether the patient has died or survived since then. For this reason, censored patients are classified neither as survived nor as died on any given day. We simply deduct them from the number of patients alive at the next time-point. When there are censored patients at the same time as patients that die, we deal first with patients that die. Then we add a new line, mark it with a little “+” right after the time count and denote the censored patient(s) by taking them off the count of patients alive at time *t*.

****

At time *t* = 6 and *t* = 9, we need to subtract one person from the risk set, the number of patients at risk of death. For *t* = 6, for instance, 4 people enter that time period, but one drops out, leaving 3 to go forward to *t* = 7, which means that at the time of the next death, *t* = 9, the proportion of patients surviving is (3 - 1) / 3.

At times when no one dies, the proportion surviving that time point is 1, or 100%, so the (cumulative) probability of survival past time *t* in the last column is unchanged.

At the last time point, *t* = 22, there's only one person left in the risk set, i.e. only one person who we're still following up, and they then die, giving a final probability of survival beyond *t* = 22 of zero.

**Results of the interpretation of log-rank test for gender**

The log-rank test compares the survival time by gender. It’s the most popular method of comparing the survival of patient groups that takes the whole follow-up period into account. Its big advantage is that you don’t need to know anything about the shape of the survival curve or the distribution of survival times. It’s based on a comparison of the observed numbers of deaths and the numbers of deaths expected if in fact there were no difference in the probability of death between the groups (genders in this case) and uses a chi-squared test.

If the resulting *p*-value is high, there’s no good evidence of a difference between the groups in their survival times.

**The Hazard and Survival Functions**

Let *T* be a non-negative random variable representing the waiting time until the occurrence of an event (i.e. the time to the event). For simplicity we will adopt the terminology of survival analysis, referring to the event of interest as “death” and to the waiting time as “survival” time, but the techniques to be studied have much wider applicability. They can be used, for example, to study age at marriage, the duration of marriage, the intervals between successive births to a woman, the duration of stay in a city (or in a job), or the length of life of any organism.

**The Survival Function**

We will assume for now that *T* is a continuous random variable with probability density function (p.d.f.)  and cumulative distribution function (c.d.f.) , giving the probability that the event has occurred by duration *t*.

It will often be convenient to work with the complement of the c.d.f, the **survival function**

which gives the probability of making it past time *t*, or more generally, the probability that the event of interest has not occurred by time *t*. A simpler equation for the survival function is

where is the number of events at time *t* and is the number of subjects at risk of experiencing the event at time *t*. is the survival probability at one time-step before time *t*. The survival function calculates the cumulative survival probability at any time.

**The Hazard Function**

An alternative characterisation of the distribution of *T* is given by the**hazard function**, or **instantaneous rate of occurrence of the event**, defined as

The numerator of this expression is the conditional probability that the event will occur in the interval [*t*, *t* + *dt*) given that it has not occurred before, and the denominator is the width of the interval. Dividing one by the other we obtain a rate of event occurrence per unit of time. Taking the limit as the width of the interval goes down to zero, we obtain an instantaneous rate of occurrence.

The conditional probability in the numerator may be written as the ratio of the joint probability that *T* is in the interval [*t*, *t* + *dt*) and *T* ≥ *t* (which is, of course, the same as the probability that *t* is in the interval), to the probability of the condition *T* ≥ *t*. The latter is *S*(*t*) by definition. Dividing by *dt* and passing to the limit gives the useful result

which some authors give as a definition of the hazard function. In words, the rate of occurrence of the event at time *t* equals the density of events at *t*, divided by the probability of surviving to that duration without experiencing the event.

These results show that the survival and hazard functions provide alternative but equivalent characterizations of the distribution of *T*. Given the survival function, we can always differentiate to obtain the density and then calculate the hazard. Given the hazard, we can always integrate to obtain the cumulative hazard and then exponentiate to obtain the survival function.

**Expectation of Life**

Let µ denote the mean or expected value of *T*. One would calculate µ multiplying *t* by the density  and integrating, so

Integrating by parts, and making use of the fact that *−f(t)* is the derivative of *S(t)*, which has limits or boundary conditions  and , one can show that

In words, the mean is simply the integral of the survival function. The median survival time is the point on the x-axis corresponding to a survival probability of 0.5. This is the time by which 50% of the subjects had experienced the event of interest.

**A Note on Improper Random Variables**

So far, we have assumed implicitly that the event of interest is bound to occur, so that . In words, given enough time the proportion surviving goes down to zero. This condition implies that the cumulative hazard must diverge, i.e. we must have Λ(∞) = ∞. Intuitively, the event will occur with certainty only if the cumulative risk over a long period is sufficiently high.

There are, however, many events of possible interest that are not bound to occur. Some men and women remain forever single, some birth intervals never close, and some people are happy enough at their jobs that they never leave. What can we do in these cases? There are two approaches one can take.

One approach is to note that we can still calculate the hazard and survival functions, which are well defined even if the event of interest is not bound to occur. For example, we can study marriage in the entire population, which includes people who will never marry, and calculate marriage rates and proportions single. In this example *S(t)* would represent the proportion still single at age *t* and *S(∞)* would represent the proportion who never marry.

One limitation of this approach is that if the event is not certain to occur, then the waiting time *T* could be undefined (or infinite) and thus not a proper random variable. Its density, which could be calculated from the hazard and survival, would be improper, i.e. it would fail to integrate to one. Obviously, the mean waiting time would not be defined. In terms of our example, we cannot calculate mean age at marriage for the entire population, simply because not everyone marries. But this limitation is of no great consequence if interest centres on the hazard and survival functions, rather than the waiting time. In the marriage example we can even calculate a median age at marriage, provided we define it as the age by which half the population has married.

**The Cox Proportional Hazards Regression Model**

Last week you calculated survival probabilities and worked through the steps involved in the Kaplan-Meier method. This week you’ll learn about the more flexible Cox proportional hazards model. With this method you will be able to compare the survival of multiple groups of patients at the same time. With the Kaplan-Meier estimator, only univariate analysis of survival can be carried out. However, as Cox PH is a regression model, we can explore the effect of multiple covariates on survival.

Like the name suggests, the model is formulated around the concept of hazards. You’ve already seen examples of the hazard function and the hazard can vary over time. The hazard function *h(t)* is the probability of the event happening at time *t*, given that it has not yet happened.**In other words, *h(t)* is the probability of dying at time *t* having survived up to time *t.*** While the concept sounds straightforward, there’s no easy formula to compute *h(t)* by hand. If you are comfortable with formulae, you can follow this link to an article explaining the hazard function <http://data.princeton.edu/wws509/notes/c7s1.html>

An important concept involved in the calculation of the hazard is the risk set. Just like the risk of dying (or experiencing some specific event) changes over time, so too does the number of patients that are subjected to that risk change over time as people die or drop out. **The risk set at time *t* is defined as the set of patients at time *t* that are at risk of experiencing the event.**You saw this in the earlier calculations for the Kaplan-Meier method when we made a risk set adjustment for patients who dropped out. Survival analysis consists of a family of methods, and one way that they differ is in their handling of drop-outs and other issues when they define the risk set.

Usually in survival analysis, we are interested in the difference between survival curves of different groups of patients. Earlier you saw the log-rank test, which gives a *p*-value for comparing the survival curves between different groups of patients with a Kaplan-Meier plot. The *p*-value tells you nothing about the size of the difference between the survival curves, however. This is done by dividing one hazard by another to give a hazard ratio. For example, dividing the hazard for females by the hazard for males gives you a hazard ratio for females compared with males. It tells you how much more likely female patients will die than male patients (i.e. a relative risk ratio).

I’ll now introduce you to the Cox model and explain how its hazard and hazard ratios work. This week you’ll get to know the most commonly used survival analysis method for incorporating not just one but multiple predictors of survival: Cox proportional hazards regression modelling. You’ll learn about the key concepts of hazards and the risk set. From now and until the end of this course, there’ll be plenty of chances to run Cox models on data simulated from real patient-level records for people admitted to hospital with heart failure. You’ll see why missing data and categorical variables can cause problems in regression models such as Cox.

**KEY CONCEPTS**

* Define a hazard in the context of survival analysis
* Run a simple Cox model in R and interpret the output
* Select and apply appropriate methods to formulate and examine statistical associations between variables within a data set in R

**Missing data in survival analysis**

Missing data are a common problem in research. The conclusions of analyses where the data are complete can be very different from analyses with incomplete data. How do you make sure your analysis yields the correct conclusion, even though the data are not complete?

First, **you need to understand why some data are missing**. This is important, because the techniques you decide to apply depend on the reason some data are missing. Be aware that there is no statistical test telling us why the data are missing. This is done by combining reason and knowledge on how the data were collected. Something I’ve emphasised throughout this course and the previous ones in the series is that **there is no substitute for getting to know your data**. Part of this is by tabulation and histograms etc., but another key part of it comes before any descriptive analysis – **knowing how the data were generated** and the potential for missing or invalid values in each data field. Let’s now recap patterns of missingness.

We say that data are “missing completely at random” (MCAR) when the complete cases (patients without any missing values for a given data item) are a random sample of the whole dataset (all patients). One patient is just as likely to have missing values as any other patient: males just as likely as females, older patients just as likely as younger ones, and so on. This can happen when a participant didn't have time to fill out the questionnaire or some information was lost or misplaced - and none of these things happened in a systematic way. This is the easiest situation to deal with, though sadly it’s often rather an unrealistic assumption.

More often, you’ll have to deal with data that are “missing at random” (MAR). In this case, missingness can be explained by other variables for which there is full information. For example, if people with a higher education are less likely to disclose their income, then income is MAR because the chance of income values being missing depends on the patient’s education. In this situation, which is common, you can predict the missing values on the basis of another variable, so if you know their education you can predict their income well. Statistical methods exist to deal with this that are beyond the scope of this course, though I’ll list them briefly below.

Finally, data that are “missing not at random” (MNAR) are neither MAR nor MCAR. For example, you could be missing medical information on the severity of diabetes when they are too ill to see a doctor and provide that information; missingness depends partly on the diabetes status, as is the case for MAR, but it also depends on the severity of illness, which can’t always be captured. In general, data are MNAR when the missingness is specifically related to what’s missing and so the probability of the value being missing depends on unobserved variables, i.e., variables not in your data set. This is generally the most problematic type.

Now that we know what we are talking about when we say missing data, we can have a look at different methods for dealing with incomplete data. Luckily, you only need to understand the general idea and pick the right tool, as the computer will do rest of the work. Here are some of the most used techniques for handling missing data.

**Complete case analysis**

In this approach, the cases with missing data are simply omitted from the analysis. **If the data are MCAR, this will produce unbiased estimates if the sample size is still sufficiently large**. If the data are MAR or MNAR, the estimates will be biased. That’s a good reason why you need to understand the reason for the missing values. It’s tempting to just hope they’re completely random, but you need to think through the problem, run some descriptive analyses and ask the data provider if necessary and possible.

**Mean/median substitution (or mean/median imputation)**

Replace (“impute”) the missing values of a variable with the mean or median of the available values of the same variable. For example, if some male patients are missing values, then just assign them the overall mean value for the male patients who do have values. **This has the advantage of not changing the overall mean for that variable**. However, **it artificially decreases the estimated variation**. It also makes it difficult to detect correlations between the imputed variable and other variables. Hence mean/median substitution always gives biased results and is not recommended.

**Multiple imputation**

Missing variables are assumed to be MAR (or MCAR) and are imputed by drawing from a distribution. This is done multiple times and yields multiple different completed datasets. Each of these datasets is analysed, and the results are combined into a single overall result. **Multiple imputation has been shown to yield unbiased results for MAR or MCAR data**. It can be done in R.

**Maximum likelihood**

**This approach also gives unbiased results for MAR (or MCAR) data**. Data are assumed to be normally distributed with a certain (multivariate) mean and variance. Observed data are used to compute the mean and variance, and missing data are drawn from the resulting normal distribution. We draw many times from the distribution **until the mean and variance of the completed data are as close as they can get to that of the observed data**. Fortunately, you don't have to do that yourself. There are many packages that can do that for you in R!

You may have noticed that I’ve not suggested any approach for MNAR data. This is because **MNAR data need to be handled on a case-by-case basis**.

<https://www.ncbi.nlm.nih.gov/pubmed/17538078>

<https://www.ncbi.nlm.nih.gov/pmc/articles/PMC3668100/>

Having run the Cox model with just one predictor and become acquainted with its code and output, you’ll want to move on to include multiple predictors in a multiple Cox regression model. As you saw with ethnic group, which had both missing values and some small categories, including one for “unknown”, leaping in and running the model without looking at the variables is hazardous.

You’ll extend the simple Cox model to the multiple Cox model. As preparation, you’ll run the essential descriptive statistics on your main variables. Then you’ll see what can happen with real-life public health data and learn some simple tricks to fix the problem.

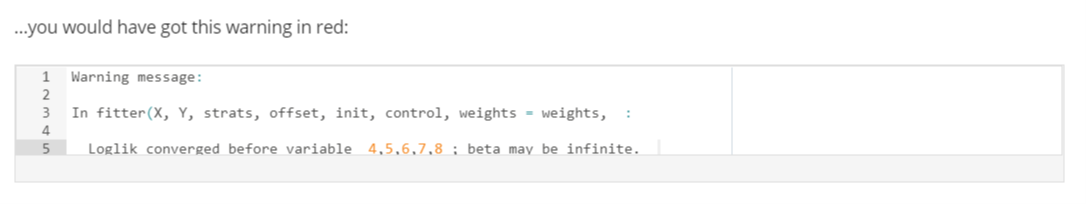
**KEY CONCEPTS**

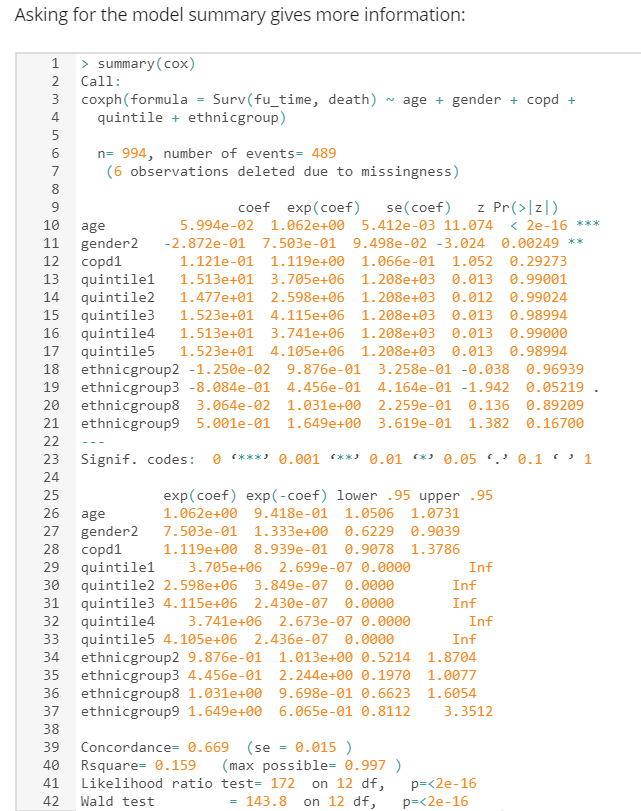
* Explain and run a multiple Cox model in R and interpret the output
* Assess the potential effect of correlated variables on modelling
* Describe a given data set from scratch, including data item features and data quality issues, using descriptive statistics and simple graphical methods as a necessary first step for more advanced analysis using R software
* Test for non-convergence in a regression model and fix the problem
* Recognise that approaches other than Cox PH exist for survival analysis

This next activity does essential preparation for running a multiple Cox regression model.

Earlier you loaded the data set for the course and ran Kaplan-Meier analysis and then simple Cox regression, but you only looked at the outcome variable – death – and a single predictor – age and then ethnic group. The latter had some missing values. Now we want to incorporate more variables into the Cox model, so we need to summarise each of them first, to see if they too have any hidden traps.

If you took the earlier courses in this specialisation on linear and logistic regression, you'll be familiar with the concept of a "multiple regression model" - it's just an extension of a simple regression model to incorporate multiple predictors. When you have a choice of predictors to include, things get complicated - later in this course, you'll look at why this can be a problem and what to do about it. For now, let's just say you want to fit a model with five predictors because **you have good prior knowledge that they're important**. Once you've decided which predictors to try, it's very easy in R to enter them into a model, whether that's Cox or any other kind of regression model.





This is a disaster. The coefficients and particularly the standard errors for quintile are all huge. A couple of the standard errors for ethnic group are a bit high, but the main problem is quintile. It seems to have infinitely wide CIs. Why?

Only four patients have quintile zero. This means invalid quintile, for instance when the postcode (zip code) can’t be mapped to a small geographical area and therefore to a socio-economic status measure. Four patients in a category can sometimes be enough to get the model to work, but there’s another problem. Of those four patients with quintile zero, not one died. That itself might not be a problem, but we’ve let R choose the reference category by default, and it’s chosen quintile zero. All the other five hazard ratios are relative to this tiny group of patients in which no one died. It’s not surprising that the algorithm couldn’t come up with sensible HR estimates. Remember, R uses the lowest integer as the reference group if the covariate is encoded or discrete.

How can you fix this?

**1) Change the reference category**

By default, R sets the first (or lowest) category to be the reference. With quintile taking on values from 0 to 5 and 0 having very few patients in it, 0 is an awful choice of reference category. The first thing to try is to set the reference one to something else. As quintile measures socio-economic status or deprivation, and you're usually interested in the effect of lower status (or greater deprivation in UK terminology) compared with higher status, it makes sense to set higher status, quintile = 1, as the reference category.

**2) Combine categories**

If that doesn’t work, then consider combining categories if it makes sense to do so. Quintile zero is an artificial category meaning that the patient’s postcode (zip code) or other small geographical area identifier was missing or could not be linked to the national socio-economic status file, i.e. the patient’s status is unknown. These patients are often different from the others, e.g. they are from overseas or are homeless, and that’s why they have no postcode or geographical area. It doesn’t make too much sense to combine them with any of the other five categories. However, in this case, this quintile zero group is so small compared with the other categories that combining them will have a negligible impact on the results.

**3) Exclude the patients**

This is the best option if combining categories doesn’t make sense and there are only a few patients in the problematic category.

**4) Drop the offending variable**

This will be the best option if combining categories doesn’t make sense and if there are too many patients in the problematic category for us to be comfortable dropping them all.

In practice, you may have to trade off one of these concerns against another, for example having to choose between combining categories that don’t fit well together and dropping an important variable from the model. **How much information is being lost or potentially distorted with each option?**

**Remember that the problem of non-convergence can happen in any kind of regression and that these simple tricks can also work there.**

**Alternatives to Cox regression**

In this course, I’m covering the two most commonly used survival analysis methods, Kaplan-Meier and Cox regression. These are popular largely because they are easy to run in standard software but mainly because you don’t need to make any assumptions about the shape of the baseline hazard function, the specific way that risk changes over time. There are, however, several other important types of methods for analysing such data. The example you’ve been using in this course concerns trying to predict mortality following admission for heart failure. Cox doesn’t care about the distribution of survival times or what the baseline hazard function looks like. Therefore, it’s called “semi-parametric”: it has some parameters – those of the predictors – but it has no parameters to describe the hazard function for patients with a value of zero for the predictors (i.e. patients with age zero and all the reference categories for the categorical variables). For completeness, the simple proportion alive at a given time point and the Kaplan-Meier estimate are examples of non-parametric survival analysis.

However, making assumptions about the shape of the baseline hazard function – adding parameters to the model to describe the shape, making the model “fully parametric” – can lead to better prediction. More accurate prediction of a patient’s survival time or risk of death within a given timeframe is vital for enabling the patient and his or her doctor and clinical team to make decisions regarding treatment. Risk models can put patients into, for example, low-, medium- or high-risk in a process called risk stratification, and high-risk patients can be offered different treatment plans from low-risk ones. It may be that you can do better than the Cox model in terms of risk prediction for a given data set and patient outcome. **The Cox model was developed to look at the effect of covariates on the baseline hazard function rather than to estimate survival times**. A fully parametric model can help here, especially if the Cox model assumptions are violated.

There are several such fully parametric models such as Weibull, exponential, log-normal, and log-logistic models, where the baseline hazard function has to be specified. The Weibull distribution is used widely in medicine because of its flexibility: its hazard function can be increasing, decreasing, or constant over time. A special case of it is an exponential distribution, which is simple because it has only one parameter. This is because the hazard function is constant when the survival time is exponentially distributed. If you want a hazard that increases and then decreases over time, try either the log-logistic or the log-normal.

There are further extensions to the basic survival analysis approach, such as allowing for the fact that the values of some predictors change over time (Cox can deal with this) and handling multiple events (patient outcomes) in the same model. This is useful for disease recurrence, for example. Also, like so many statistical methods, survival analysis can be run in a Bayesian framework. Bayesian analysis is, in general, more complicated but very powerful. It involves mixing your data and your prior beliefs about what is related to what and deriving probabilities that something is true. What I’m teaching on this course and throughout this series of courses within the specialisation is called classical or “frequentist” statistics. In the classical framework, there’s no formal use of prior knowledge in the underlying maths. The answer you get is completely driven by the data. There’s a philosophical as well as a mathematical difference between the two, and much has been written about it – it’s a huge subject, way too big to go into here. It’s often claimed that there are two rival camps, but (happily) it’s also often claimed that there are no such camps and that many people, including me, use both methods (which often give similar results anyway in practice).

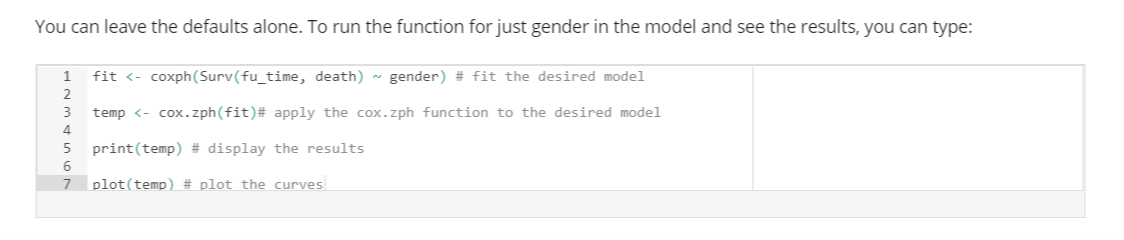
**The Proportionality Assumption**

In this final part of the course, you’ll learn how to assess the fit of the model and test the validity of the main assumptions involved in Cox regression such as proportional hazards. This will cover **three types of residuals**. Lastly, you’ll get to practice fitting a multiple Cox regression model and will have to decide which predictors to include and which to drop, a ubiquitous challenge for people fitting any type of regression model.

**KEY CONCEPTS**

* Evaluate the model assumptions for Cox regression in R
* Apply a simple way to fix the problem of proportionality assumption violation
* Apply and critique simple ways to deal with missing values in a predictor
* Describe and compare some common ways to choose a multiple regression model

The proportionality assumption can be checked informally by plotting the hazards. If the assumption is met then the hazard lines will be roughly parallel to each other - note that that's only true when they're plotted on the log scale, i.e. if you take the natural logarithm of the hazards or plot them on axes on the log scale.



Technically speaking, the function cox.zph() correlates for each predictor the corresponding set of **scaled Schoenfeld residuals** with time, to test for independence between residuals and time. You don’t need to know what any of that means to do the test, but some people prefer to know the technical details of things. If you took the courses on linear and logistic regression in this series, you will have come across the term “residuals”. In those types of regression, they measured the difference between the model’s predicted values and the actual values from the data. Cox regression also generates residuals, and Schoenfeld are one type mentioned in the video.

A *p-*value greater than 0.05 from the cox.zph test confirms proportional hazards. **If the plotted line is relatively flat, the effect of the covariate changes little during the follow-up period**. That’s good news. Schoenfeld residual plots test proportionality of hazards.

**When you have a predictor with few categories, you can also use our old friend the Kaplan-Meier plot as an informal visual check**. If the predictor satisfies the proportional hazard assumption, then the graph of the survival functions versus the survival time should yield parallel curves. This method does not work well for continuous predictors or categorical ones with many levels because the graph becomes too “cluttered”.

**Deviance residuals** are transformations of Martingale residuals and help you look for **outliers or influential data points**. You can either examine the influence of each data point on the coefficients or plot the distribution of the residuals against the covariate:

res.cox <- coxph(Surv(fu\_time, death) ~ age)

ggcoxdiagnostics(res.cox, type = "dfbeta", linear.predictions = FALSE, ggtheme = theme\_bw())

It’s also possible to check outliers by visualizing the deviance residuals, which are normalised transformations of the Martingale residual and **should be roughly symmetrically distributed about zero with a standard deviation of 1**. If you remember the normal distribution, then 5% of observations are more than 1.96 standard deviations from the mean. If the SD is 1, then only 5% of observations should be bigger than 1.96 or more negative than -1.96. If you have more than that proportion, then **your model doesn’t fit the data as well as it should**, and some observations are a problem. This is just the same issue as with the other types of regression. The maths behind the calculation of the residuals is different, mostly because of the censoring, but we don’t need to worry about that.

* Positive values correspond to individuals that “died too soon” compared with expected survival times.
* Negative values correspond to individuals that “lived too long” compared with expected survival times.
* Very large or small values are outliers, which are poorly predicted by the model.

res.cox <- coxph(Surv(fu\_time, death) ~ age)

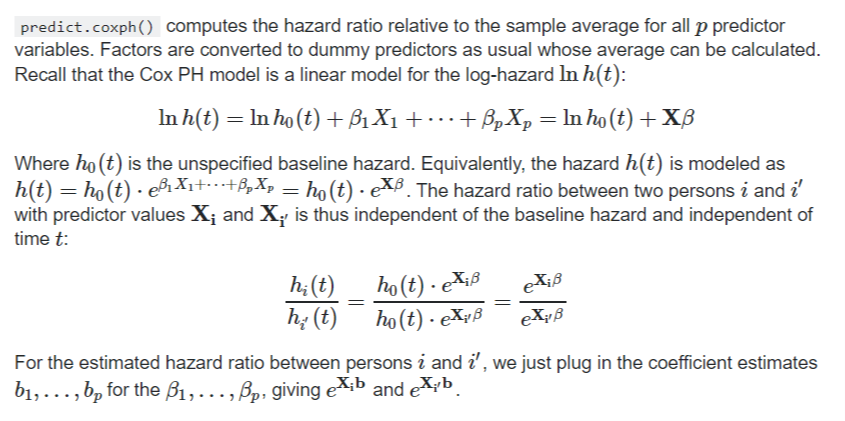
ggcoxdiagnostics(res.cox, type = "deviance", linear.predictions = FALSE, ggtheme = theme\_bw())

Another issue is whether any continuous variables that you assume to have a linear relation with the outcome do have a linear relation. If you fit age as a single term in the model, then that’s what you’re assuming. The **Martingale residual** is used to test this assumption:

ggcoxfunctional(Surv(fu\_time, death) ~ age + log(age) + sqrt(age))

Martingale residuals may present any value between minus infinity and 1) and have a mean of zero:

* Martingale residuals near 1 represent individuals that “died too soon”
* Large negative values correspond to individuals that “lived too long”
* The plots should give you nice straight line if the assumption is valid



*If you have a stratified model, the comparison in predict.coxph() is against the strata-averages (means), this can be controlled via the reference option that is explained in the help page.*

*Very clear answer. Thank you. I have a question about using the mean of a skewed continuous covariate for the reference average hazard calculation (by strata or overall sample). Wouldn't an extreme positive skew on, for example, age make any reference mean age a poor measure of central tendency? A median would be a better statistic unless skewed covariates can be transformed to normal?*

**Suppose your test for proportional hazards gives you a clear suggestion that the assumption isn't met. What should you do?**

To answer this, you need to think about what having non-proportional hazards means. If the relation between males and females regarding their risk of death changes over time, it could mean, for instance, that males have a higher risk of death early on during the follow-up period but at some point the relation changes so that females have a higher risk of death. One way of putting this is that there is a statistical interaction between gender and time. The model is short of a coefficient. If you add a coefficient for this interaction, which allows for the difference in risk by gender to change over time, then the problem would be solved.

Trying this interaction term in the model and testing whether it is statistically significant is in fact another way of testing the proportionality assumption. If this interaction term is not statistically significant, then it follows that the assumption is valid. As is usual with any kind of regression, Cox included, you should do the statistical tests – i.e. get the p-values – but also do the plots. Some kinds of non-proportional relationships and other assumption violations can’t be detected just from a p value.

Let’s go through how to include this interaction term and test whether it’s statistically significant. For mathematical reasons, you can’t just include the follow-up time itself as part of the interaction but instead need to transform it. The easiest way to do this in R is via the “tt” function (short for “time transform”):

This output agrees with the earlier approach and says that the interaction between gender and (transformed) time is not statistically significant, i.e. there’s no apparent violation of the proportionality assumption. Again, good news. The p-value from this approach (about 0.5) isn’t the same as that from the earlier one because the methods are different, though it’s always nice when they give the same message!

So, if the assumption is violated, then one option is to include this interaction. If the p-value is low but the hazards are proportional for most of the follow-up period, then that suggests another solution: divide the survival analysis into two time periods. You can fit one model when things are fine, i.e. when the assumption is valid, and another model to cover the later follow-up period when the assumption is not valid. This second model may need an interaction term, but the first one won't.

There’s also a third simple way of dealing with the problem: stratify the analysis by the variable that’s causing the problems. If it’s gender, for instance, then just fit separate models for males and females. The drawback of this approach is that it’s no longer possible to compare the effect of each gender on mortality.

**Model selection methods: how to choose your predictors**

This was covered in detail in the Logistic Regression for Public Health course – similar principles apply to any type of regression, including Cox models. There, I explained some common ways of choosing predictors for a multiple regression model and that two such ways – **forwards selection and stepwise selection – were simply too awful to contemplate using**. A third common way**, backwards elimination, does sometimes work okay**. **While it’s always good to make use of a priori knowledge from the literature and experts in the field, this isn’t always of enough help, particularly when you have a lot of possible variables**. Less often you’ll have a good deal of a priori knowledge and therefore many predictors that have been found to be associated with the outcome. In that situation, it can be useful to apply backwards elimination to the model with all these chosen predictors in order to reduce the size of the results table for presentation.

**How to apply backwards elimination**

Here are the steps:

1. Fit the model containing all your chosen predictors – either all your a priori ones or all your available ones (if your data set isn’t too large); beware of the *n* *predictors* > *n* *observations* problem;
2. Store all the coefficients from that model;
3. Remove in one go all predictors whose *p*-value is above the pre-set threshold, typically the usual 0.05 (in a variant of this, you remove the predictor with the highest *p*-value and refit the model, repeating steps until all the predictors have *p*-values above the chosen threshold);
4. Compare the coefficients for the remaining predictors with their coefficients from the original model;

**Checks to make when using backwards elimination**

**If the coefficients haven’t changed much from the original model, then you now have your final model**. You can go ahead and check the residuals (Schoenfeld, deviance and Martingale) and other model assumptions (e.g. proportional hazards). **If, however, you have a predictor whose coefficient has changed noticeably, then you need to find the variable(s) that you have removed that are correlated with this affected predictor**. You can do this by trial and error, so add back in one of the removed variables at a time **until the affected predictor’s coefficient is back to its original value**. **When that happens, you’ll need to keep the removed variable in the model**.

For example, suppose that blood pressure was retained (original model HR=1.30, *p* = 0.002) but cholesterol was removed because it was not statistically significant (original model HR=1.05, *p* = 0.155). Then you removed cholesterol from the original model, and the HR for blood pressure changed from 1.30 to 1.50. You consider that a big enough change to worry about (see below for more on this). You add cholesterol back in, and the original HR for blood pressure is restored. You now need to keep both blood pressure and cholesterol in your final model. **Such correlation between variables is a big reason why stepwise procedures are so unreliable**.

**NB**: how big is “big enough to worry about” is arbitrary. Anything less than 0.05, e.g. a change from HR=1.30 to HR=1.34, is not big enough in my opinion. It’s up to you to decide this. **It largely depends on how the results are going to be used**, e.g. in a risk calculator for clinical decision-making, perhaps in a national screening programme, or for an epidemiological study of risk factors. In the former, where people can be invited for screening for some disease based on their estimated risk of developing that disease, using the coefficient of 1.30 instead of 1.50 can greatly affect the number of people invited. In the latter, however, such a difference won’t be of such importance, especially if all we do is take the table of hazard ratios and *p*-values and say, “these are the predictors of the outcome, whereas these other factors are not”. The estimated size of the relation (the HR) is of secondary importance.

**Conclusion**

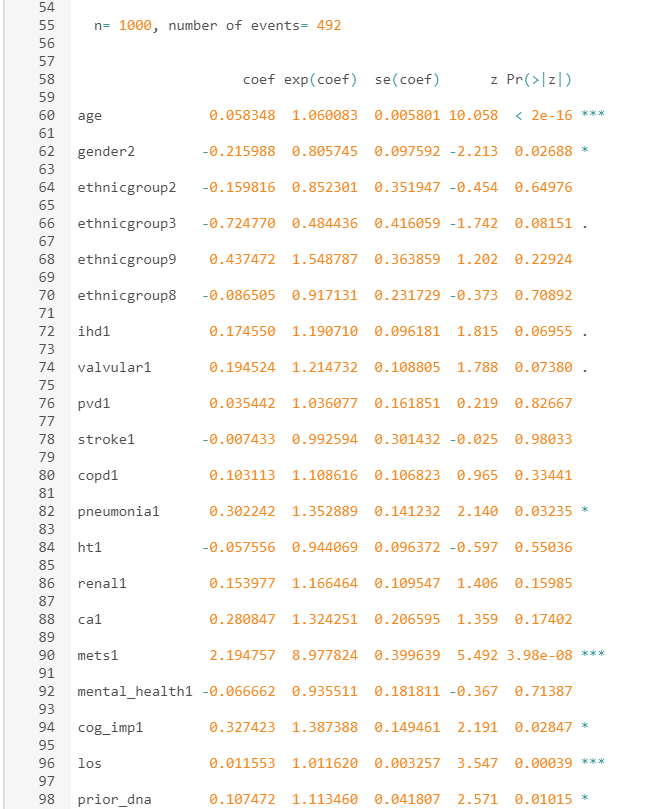
The question of how to choose the predictors in a regression model, be it linear, logistic, Cox or other type of regression, is a huge one when the number of possible predictors is bigger than a handful or two. As this is an introductory specialisation, I've only talked about a few approaches.

**Preparing the data for Cox regression**

For those variables that you haven’t yet used, the first thing to do is read the documentation and summarise their distribution. Most of the variables are simply binary flags to indicate the presence or absence of a given comorbidity. One variable (cognitive impairment) needs to be created from a combination of two existing flags.

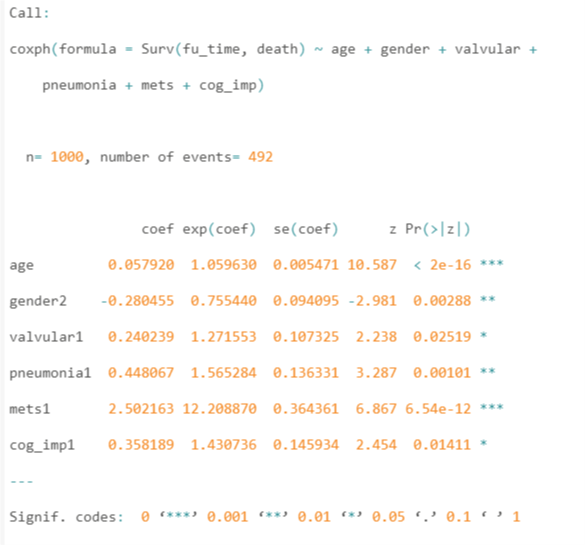
Once you have a sense of how many missing or weird values there are, you can decide what to do about them. In this data set, missing values are not a big problem.

The next question is how to include each variable in the model. For binary variables, this is easy, and you will generally want the zero (e.g. no cancer, no pneumonia) to be the reference category. For the variables that can be considered continuous although they are recorded as integers – age and missed prior outpatient appointments – you should plot their relation with the outcome and decide whether you can just assume a linear relation or whether something more complicated is needed (Martingale residual can be used for time-to-event data?). Here, we’ve decided for reasons of simplicity that linear is fine, but I want to stress that in general you should always consider something more complicated if that’s what the data tell you.

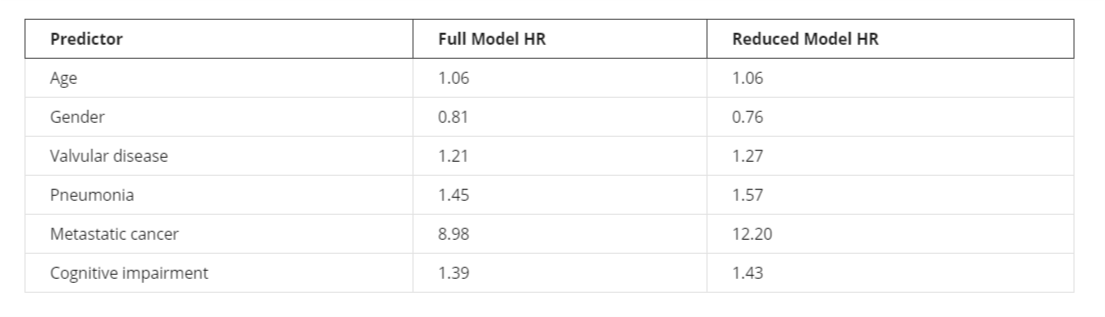


The first thing I looked at with this output was the standard errors, which are all comfortably low. Not tiny, as in less than 0.1, but certainly not big enough to worry about.

Then for this exercise you’re interested in the p-values to see which predictors you could drop to make the final table of results much more digestible. Only six have p-values below the conventional threshold of 0.05: age, gender, valvular disease, pneumonia, metastatic cancer and cognitive impairment. Two more – renal disease and ischaemic heart disease – have dots by their p-value, meaning that they’re just about the 0.05 threshold, but just above is still above. Once you’ve chosen an a priori threshold, you have to stick to it. So what happens when you keep these six and drop the rest?



As the full model converged without problems, this simpler model will as well, and it did. Next, you need to compare the two sets of coefficients. The table below does this, using the hazard ratios, i.e. the numbers from the “exp(coef)” bit of the output.



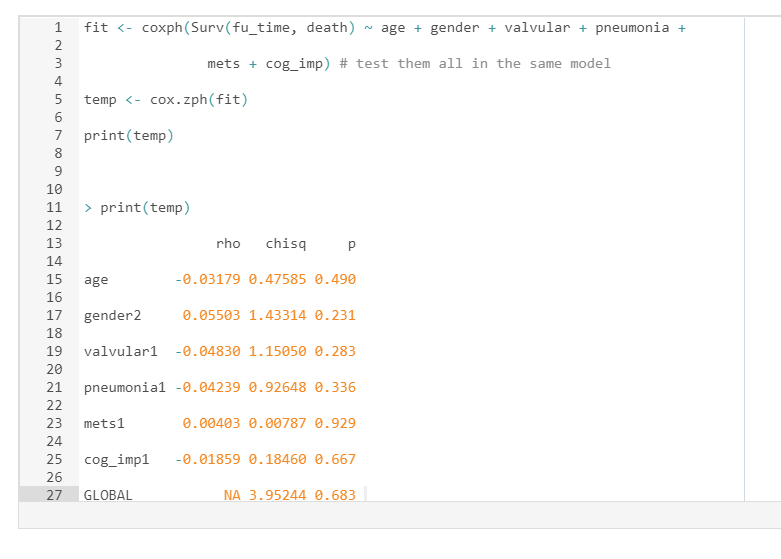
**What do you think of this comparison?**

The HRs for age, valvular disease, pneumonia and cognitive impairment are really similar, and the HRs for gender are also pretty close. That leaves metastatic cancer, whose HR went up from 8.98 to 12.20. That looks like a big increase, so should you worry? I think it depends on how the results are going to be used. If you just want to say which are the significant predictors and the approximate strength of the relation with mortality, then this is fine. If the HR is 9 or 12 it doesn’t matter – it’s a really strong relation. And there’s more evidence you can look at. When you did your descriptive statistics earlier on – I presume you *did*do them all – then a simple tabulation and then cross-tab between cognitive impairment and death will have shown this:

So only 79 patients had cognitive impairment, of whom 59 (74.7%) died. If you were doing logistic regression, you could convert this into an odds of death, giving 59 deaths divided by 20 non-deaths = 59/20=2.95. The impaired patients are three times more likely to die than they are to survive. For those without such impairment, the risk of death is 47% and the odds are 433/488=0.89. That’s pretty high. When logistic regression gives you an odds ratio for impairment, where their risk of death is 74.7%, or 3 in 4, the odds ratio will be high. When you have other predictors, the odds ratio will change, and because the odds for those without cognitive impairment are so high, it doesn’t take much for the odds ratio for those with the impairment to change a lot. If you recall the table of probabilities, odds and odds ratios from the course Logistic Regression for Public Health, odds ratios can get very high for only small changes in the underlying probability. It’s the same principle for hazard ratios from survival analysis. A change in the model (the removal of various covariates) that has little effect on HRs that are close to 1 – for example for age and gender – has a big effect on HRs that are far from 1, in this case much larger than 1. In summary, the change from an HR of 8.8 to 12.2 is not surprising and not important in this case.

**Testing the proportionality assumption on the remaining variables**

You're left with six variables to test. Here are the results:

  
Nice high p-values, so all’s well on that front.

**Why this final model differs from that in the report**

Even though similar data and variables were used in the report and to create your model, you got different results from the report. Why?

Various reasons:

1. The data were similar but not the same
2. The original set of variables was similar but not the same
3. The statistical methods and outcome were different
4. The size of your data set was much smaller than in the report

Let’s consider each of them. The data used in the report were a combination of Hospital Episodes Statistics (HES) and other, aggregate-level information. The data used for this course originated from HES but were put through a simulation macro in another stats package, SAS, so that the correlation structure and distributions for each variable matched those of the original HES but none of the original patients’ data were retained. So your data look like HES, walk like HES and talk like HES but they’re not HES, so patient confidentiality has been protected.

Secondly, the original set of variables in the project was larger than what you used. Information on intensive therapy unit use was something I mentioned earlier, but the inclusion of hospital-level and other information also had an impact on the p-values and standard errors of the other variables in the model.

Thirdly, the report used logistic regression with an outcome of mortality within one year of the admission, whereas in your survival analysis, the follow-up period was often much longer than a year. This will have an impact on the observed relations between each predictor and the outcome.

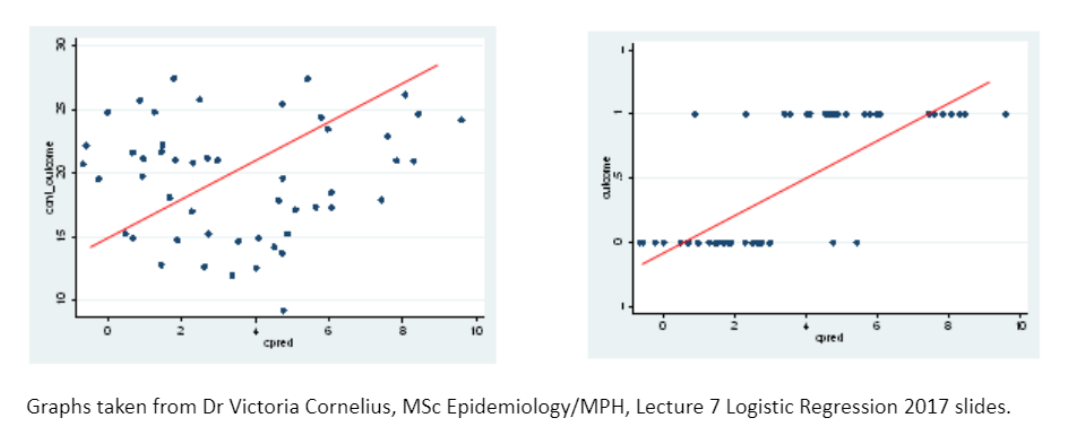
The main reason, however, is simply the size of the two data sets. The original report took two years of HES data, nearly 80,000 patients with heart failure, whereas for this course I generated a much smaller subset. That means you had a lot less statistical power to detect significant relations between a predictor and the outcome. Your final model contained far fewer variables than in the report and has without doubt missed some important predictors.

All this means that people studying the same problem – the effect of patient characteristics on mortality after admission for heart failure – can get different results if they don’t use the same data or methods. A thousand patients sounds like a lot – and it is a lot if you had to go out and recruit them yourself from scratch one patient at a time – but it’s still not enough to be able to pick up all the risk factors.

**Logistic Regression**

**Why does linear regression not work with binary outcomes?**

**Binary outcomes only have two values** and are a discrete/categorical/binary variable type.The example we are using throughout this course is diabetes, where individuals either have diabetes or they don’t. For our regression model, we could code this outcome so that individuals with diabetes = 1 and those without diabetes = 0. If we just ran a linear regression model with this binary outcome and one continuous predictor variable, then the model will plot a straight line through these points just as we have seen with simple linear regression in the course on Linear Regression for Public Health.



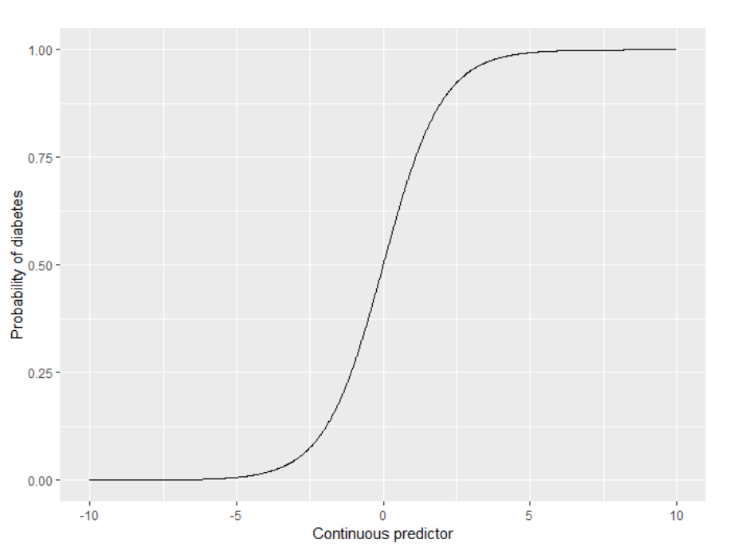
The graph on the left shows the relationship between the continuous predictor variable (cpred on the x-axis) and some continuous outcome variable on the y-axis. It shows that the predicted values from the linear regression model (red line) are somewhat reasonable for the continuous dependent variable, even if the model does not explain the relationship very well because lots of the points are far from the red line. In fact, the plot on the left looks like non-constant variance could be present as the data points fan out at lower values of the predictor variable. However, the graph on the right clearly shows that the linear model does not fit the data well at all when the outcome is binary. The **predicted values often correspond to impossible values of the outcome**, i.e. values other than 0 or 1. It just doesn’t make sense.

**For a binary outcome, we are generally most interested in modelling the proportion of individuals with the outcome of interest**, i.e. the proportion of individuals with diabetes in our example. This is **equivalent to the probability of an individual having diabetes**. Although probabilities are continuous variables, they can only take values from 0 to 1. However, as we have seen in the graph above (right), linear models will predict values below 0 and above 1. Luckily, **we can transform our variable of interest into one that can be modelled in a regression equation using something called a** **link function**.

**As its name suggests, a link function describes the relationship that links our variable of interest to the variable that we use in our regression equation.** It's a mathematical trick. The link function that’s most often used for logistic regression is called the **logit**. Instead of directly modelling the probability, **we model the logit of the probability**. The logit of a probability *p* is equivalent to the natural logarithm (log) of the odds

As the link equation above shows, to get back to *x*from its natural log (*y*), you raise *e*to the power of *y*. This ‘**anti-log**’ transformation is known as exponentiating. With the logit link function, we can go from **probability of outcome** to **odds of outcome** to **log odds of outcome** and back again.

The reason we model the log(odds) rather than just the odds as the outcome variable is because it can take any value from minus infinity (when *p* = 0) to positive infinity (when *p* = 1). Odds, on the other hand, can only take positive values. **Using the log(odds) as the outcome variable means that we can run a regression model in a similar way to normal linear regression with a continuous variable**, and still ensure that the predicted values for probabilities are between 0 and 1 (as we can convert predicted log odds back to probabilities via the logit function). See the **sigmoid function** below.



Logistic regression involves modelling the odds rather than the probability (although we can convert to probabilities after the fact). To be precise, **it models the log(odds) and can take negative and positive values which is necessary to make the mathematics work**. Therefore, we can treat it like a linear regression problem as we are not bound by 0 and 1. The equation can be written just like a simple linear regression

Where *log(odds)* is the dependent variable, *β0*is the intercept term, *β1*is the slope or coefficient estimate, *x* is the value of the predictor, and *e* is the residual error.

A null model containing no predictor variables can tell us the log odds of having an outcome such as diabetes. However, we don't just want to know the log odds of something happening, like having diabetes, we want to know how the log odds varies by some patient factor. Let’s use age as an example. Are older people more or less likely to have diabetes than younger people? In logistic regression, this is assessed by comparing the log odds of having diabetes in older people with the log odds of having diabetes in younger people. **Dividing the former by the latter gives the log odds ratio**.

Happily, we can take the antilog of the log odds ratio, a procedure called exponentiating, to get the **odds ratio** which is much easier to interpret. This is just one odds divided by another odds. For example, if we divide the odds for older people by the odds for younger people, and the resulting odds ratio is greater than one, it means that older people are more likely to have diabetes than younger people are. That's true.

We've seen that probability and odds are not the same thing, and the maths of logistic regression works on the log odds scale. Happily, the easy trick of exponentiating the output from a logistic regression model gives us odds ratios, which can be interpreted much more easily.

**Odds ratios and examples from the literature**

As I explained briefly in the video, to get the underlying mathematics to work when our outcome is binary and not continuous, the standard approach with logistic regression is to convert the binary outcome into log odds. Let’s look at that in a bit more detail.

While probabilities are the most intuitive way to model binary variables, if we model the probability as the outcome variable, we will end up with impossible predictions from the model (i.e. predicted values below 0 or above 1). The reason we model the log(odds) rather than the odds as the outcome variable is because odds can only take positive values, and therefore this model would also give impossible predictions. **The log(odds) can take any positive or negative values, and therefore all predictions from the model are possible**.

Logistic regression is a very common tool for public health studies, as many healthcare-related outcomes are binary measures.Therefore, you are very likely to come across odds ratios in the public health literature.

**Revision Questions**

**Suppose you have an outcome variable with three different values. The following is true.**

*To run logistic regression, it’s essential to combine two of the values so that the outcome is binary.*

*You can only perform logistic regression if the outcome variable is binary. Note that although combining two categories would make things simple to analyse, this isn’t always the right thing to do. You need to think carefully about the implications and information loss associated with merging categories.*

*The decision to combine two of the categories should be informed by factors such as how many patients have each value, how sensible it is to combine the categories, and what the potential impact on the study’s conclusions would be.*

**What is the mathematical quantity that is modelled in logistic regression?**

*The natural log of the odds (via the logit link function).*

**Suppose that the odds of having diabetes in an individual with BMI under 25 is 0.2 and the odds of having diabetes in individuals with a BMI of 25 or over is 0.6. What is the odds ratio of having diabetes in those with BMI of 25 or over versus those with BMI under 25?**

*0.6/0.2 = 3 (we conclude that those with a BMI >=25 are 3x more likely to have diabetes).*

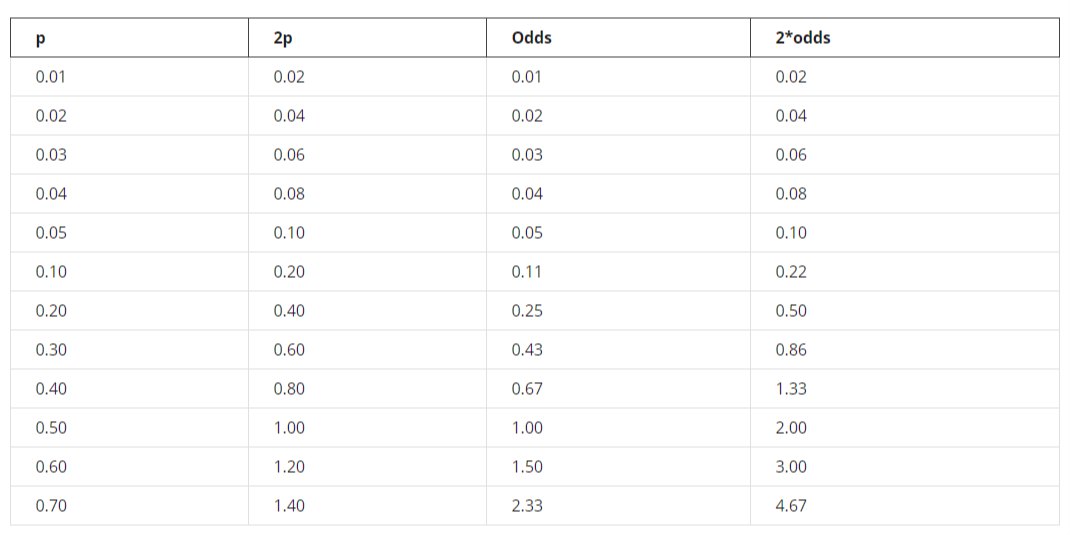
**The odds of having diabetes is lower in those with normal blood pressure than those with high blood pressure. What will the odds ratio be when dividing the odds of diabetes in people with normal blood pressure by the odds of diabetes in those with high blood pressure?**

*It will be below 1.*

**If the probability is 0.5, what are the odds?**

*1 (evens or 50/50).*

**As with all types of statistical analyses, you need to be wary of spurious results.** If a public health study presents huge odds ratios, you need to interpret these results with caution. As you can see from the table below, **as the probability increases, the odds increase in a non-linear fashion**. This means that a relatively small increase in probability can lead to a large increase in odds.



**Data Preparation for Logistic Regression**

Explore candidate variables chosen using domain expertise and statistical thinking. Summary statistics and cross-tabulations with the outcome are good to carry out as this is a binary logistic regression problem. We could also group continuous variables, such as age, and perform cross-tabulations as with other categorical variables. It is a good idea to tabulate the outcome variable.

**Logistic regression can handle categorical and continuous variables**. You do not need equal numbers in each category but **small numbers in a category can be problematic**. Continuous variables **do not need to be normally distributed**, but **we assume the relationship with the binary outcome is linear**.

Assumptions of logistic regression:

* A linear relationship between the log(odds) and the covariates
* Independent observations
* No multicollinearity
* Large sample size

Your predictors can be categorical or continuous. If categorical, you do not need an equal number of observations in each category, but categories with very small numbers can cause problems. If continuous, they do not need to be normally distributed. If variables are continuous, they do not need to be normally distributed. For a continuous variable or one that you are essentially treating as continuous, for example a year of age, you are assuming that the relationship between the variable and the outcome is linear.

For example, suppose you want to know how diabetes risk varies with age, and you have age in whole years rather than its categories. If you plot the rate of diabetes by age, you are assuming, and R is assuming, that the diabetes rate changes by the same amount for every one unit increase in age, whether it goes up with age, goes down with age or is flat. No curve is allowed. The relation is linear. **In practice, the relationship is unlikely to be perfectly linear so you need to ask if the model will be useful given an approximately linear relationship**.

It's important to test that your data fit this assumption by plotting the data first and then fitting a model. Don't just hope for the best. If the relationship on your plot looks rather more curved than straight, then maybe a line isn't a good approximation. In that case, you will need to try some other shapes, for instance, by adding a squared term to the model or a spline. If your single predictor is age, then this would mean including not just a term for age but also a term for age squared and testing whether that squared term is statistically significant via its *p*-value.

**Simple logistic regression: how to run a model with only one predictor**

The simplest model we can fit is one with no predictors whatsoever in it. This is called the empty or **null model** and is rarely useful. **It has the assumption that everyone has the same odds of having the outcome** (diabetes in our example).

glm(dm ~ 1, data = df, family = binomial (link = logit))

The “1” is just R’s way of saying that there’s only an intercept term in the model. To get the output, though, you need the “summary” command as well. To do this, you need to make an R object out of the model. I’ve called this object “mdl”. We can then summarise it:

mdl <- glm(dm ~ 1, data = df, family = binomial (link = logit))

summary(mdl)

...you see that there are 60 1s and 330 0s, which is good because there were 60 yes’ and 330 no’s in the “dm” vector. **It’s important to know that R is modelling the log odds of dm=1 and not the log odds of dm=0.**

Much more useful than the null model above is to see how the chance of having diabetes depends on one or more predictors. The next simplest model is one with one predictor. Let’s first look at gender, which in our data set is binary: male or female (the reference level will be 0 or female).

We’ve already told R that gender is a factor (i.e. a categorical variable).

mdl <- glm(dm ~ gender, data = df, family = binomial (link = logit))

summary(mdl)

This means we are saying that the log odds of having diabetes differs by gender alone. To include a continuous variable in the model instead, such as age (note that we are treating age as continuous, just rounded down to the nearest birthday), then the code is similar.

**R will know to fit age as a continuous variable because you haven’t told it that it’s categorical**. If the age vector is numeric or integer atomic type, then R will treat it as a continuous variable.

It’s straightforward to include age as a single term in the model, but remember what I said in the video about assuming a linear relationship with the outcome? More precisely, this assumes that the relationship between age and the log odds of having diabetes is linear (more on this in detail in the next section). Is that reasonable? **The easiest way is just to plot one against the other**.

The most interesting section of the model summary is the list of coefficients, but first there’s another bit of info that R has sneaked in that is worth noting. It’s in brackets but it’s important. **13 observations were deleted due to missingness**. For 13 of our patients, we don’t know whether they had diabetes, so they’ve been excluded. Luckily, that’s not a large number and it’s not a large proportion of our sample, so we can just note that down and move on to look at the coefficients. The default behaviour in R is to skip observations where NAs are present in one or more model covariates.

**In the null model, there’s just one coefficient: the intercept.**R prints out all the coefficients on the scale on which the algorithm did its magic, i.e. the log scale in the case of logistic regression as we are modelling the log odds of having diabetes. With this rather unexciting null model, we are saying that the log odds of having diabetes is -1.7047 and that it’s the same for every patient. What does that mean? To interpret this, we first need to exponentiate it to get the odds of having diabetes.…and you’ll get 0.182 to three decimal places. If you prefer to work in probabilities rather than odds, you can use the relationship between odds and probability that we established earlier to convert. So, just divide the odds by 1 plus the odds, to give 0.182/1.182 = 0.15, or 15%. How do these compare with the raw data? If you tabulate the outcome variable, the odds of having diabetes is 60/330 = 0.182 and the probability is 60/(330+60) = 0.15, both exactly the same as from the model, which is entirely as we had expected (and hoped!).

Now let’s add age. We still have 13 observations that were omitted due to missingness. This shouldn’t be a surprise, because our descriptive statistics earlier in this course told us that 13 patients had no HbA1c readings – and hence no diabetes information – and no patients had their age missing in the data set. As you add more variables to the model, you’ll need to keep an eye on this.

Let’s go on and look at the coefficients. This time there are two: the intercept and one for age. Now, **with a predictor (age in this case) in the model, the intercept is no longer the overall crude log odds but is instead the log odds of diabetes when age is zero**. This follows from the equation for the model:

*Log odds of having diabetes = intercept + (coefficient for age \* age in years)*

*= -4.4045 + (0.0525 \* age in years)*

If age in years is zero, then we only have the intercept left. At birth, the model is saying that the log odds of having diabetes is -4.4045, which is 0.012 when exponentiated to give us the odds. If you prefer to think in probabilities, then we can convert this as before to give us 0.012/1.012 = 0.012 (to three decimal places) or 1.2%, which is pretty much the same as the odds. As you saw earlier, when odds are small, they’re similar to probabilities.

**So how do we interpret the coefficient for age?** It’s the change in the log odds of having diabetes for a one-year increase in age. A linear relationship is assumed between age and the log odds. It’s assumed, therefore, that the log odds if you’re 25 is 0.0525 higher than if you’re 24 and that the log odds if you’re 75 is 0.0525 higher than if you’re 74. One of the nice things about working on the log scale is that the difference between two log odds and the ratio of two log odds are the same mathematically:

* the log odds if you’re 25 minus the log odds if you’re 24 is 0.0525
* the log odds if you’re 25 divided by the log odds if you’re 24 is 0.0525

**When we exponentiate a coefficient estimate, for example 0.0525, we get 1.05 (to two decimal places)**. This is an **odds ratio**, which is generally what’s reported when running logistic regression models (and is generally reported to two decimal places). It’s the ratio of the odds of having diabetes if you’re 25 divided by the odds of having diabetes if you’re 24. Or that if you’re 75 divided by that if you’re 74 etc. It’s the amount by which your odds increase when you get a year older. Therefore, getting older is bad news, at least in terms of diabetes, which is what we expected.

But wait – we haven’t yet checked to see whether this is statistically significant or merely compatible with a chance result. The *p*-value for age is given in the “Pr(>|z|)” column and is tiny. R also uses an asterisk system to point out the size of the *p*-values. Age has three asterisks, meaning close to zero, so it’s not compatible with a chance result. **Age is a statistically significant predictor.**

In this data set, gender comes labelled as male and female, so the coefficient for gender here is printed as “gendermale” as the reference, female, is based on alphabetical order (i.e. female comes first). If your data set has gender coded as 1 and 2, for example, then you’ll need to refer to the documentation for that data set to see what 1 and 2 mean (1 would be the default reference unless otherwise specified). Here, the log odds for having diabetes for males is 0.0869 higher than that for females. This is also the log odds ratio for males compared with females. Again, if we exponentiate 0.0869, we get the odds ratio for males compared with females, which is 1.09 (to two decimal places). That implies higher odds for males, but we need to inspect the *p*-value, which is 0.759. That’s well above the conventional threshold of 0.05, so chance is a likely explanation for the result, and we can conclude that we don’t have any good evidence of a gender difference in diabetes odds in this sample.

While in this data set, gender is nicely labelled, it’s a good idea in general to check how R has entered gender into the model using the contrasts function. This confirms that the coefficient given in the output refers to males because males have a 1 next to them in the above output and females have a zero. The log odds for females is incorporated into the intercept (i.e. the intercept is the log odds of females getting diabetes and the log odds only changes if gender = male).

Suppose you didn’t want to compare males relative to females but instead the reverse?**How can you get R to give you the odds ratio so that it’s the odds for females divided by the odds for males?**

R will by default organise values (called levels) of categorical variables alphabetically. **Therefore, by default, “female” is the first level.** R will automatically set this as the reference group in statistical analyses, such that the odds ratios of other groups will be displayed relative to this one. Remember the table in section 1.07? The odds of having diabetes with hypertension is compared with the reference group (not having hypertension). This makes sense, as we would hope that not having hypertension would be the “default” state. The default gender is, however, more arbitrary, so it may make sense to redefine the reference group:

We can use the function relevel(x, ref = “female”) to do this. Note that the estimate is the same, except negative. This makes sense, because of the rule: **log(A/B) = −log(B/A)**

**Modelling the effects of age, gender and BMI.**

Run a model with age, gender and BMI in which age and BMI are assumed to have a roughly linear relationship with the log odds of having diabetes. **First look at how many patients were excluded due to missing values**. Normally, the next thing you’d look at is the model fit information, but I’ll cover that later as **we need more than just the default output to assess that properly**. At first glance, the fit is okay. There’s no point in interpreting the coefficients unless the model fit is reasonable.

**Next, look at the coefficients and their standard errors,** to see if there’s been some problem getting the algorithm to work or whether you've got things like categories with very few patients in them. This is an issue I'll return to in the course on survival analysis. The standard errors are all small, so that’s fine.

**Now to the interesting part.** The log odds ratio for age is 0.055 and its *p*-value is tiny, so you can conclude that age is significantly – and positively – associated with the risk of getting diabetes. If you exponentiate its coefficient we get 1.057, or 1.06 to two decimal places. This means that a one-year increase in age is associated with six percent higher odds of being diagnosed with diabetes. You ran a simple logistic regression model earlier and got a similar log odds ratio of 0.052. **The important thing about your new estimate for the effect of age is that this one is adjusted for the effects of gender and BMI**. This means you don’t have to worry about whether the apparent effect of age in your earlier simple model is in fact due to gender or indeed to BMI. However, it might still be due at least in part to things that you haven’t yet put into the model.

You’ll remember if you took the previous two courses in this series that you also need to recognise that there’s some **uncertainty**about this estimate of 6% higher odds because it’s based on a sample of patients. If you got hold of data from another sample of patients, you might not get 6% again. To express this uncertainty, you need to calculate the 95% confidence interval using the standard error. The easiest way to do this in R is with the “confint” function.

There are a lot of unnecessary decimal places, but it does the job. **For age, the odds ratio is 1.06 with 95% CI 1.04 to 1.08,** all to two decimal places as is usual reporting practice. If you like technical details, the confidence limits 1.04 and 1.08 for age (and those for the other variables and the intercept) are what’s called profile-likelihood limits. These are thought to be superior with small sample sizes, but the difference between them and alternatives such as Wald limits that R provides using the “confint.default” command is generally modest. You can try that out here for yourself.

The next variable is gender. You ran a model with just gender in it earlier, just as you did for age, and that time it wasn’t statistically significant. **Here too the *p*-value is large at 0.448, well above the standard 0.05 threshold**.Its 95% CI is wide, from 0.68 to 2.41, so this data set doesn’t tell you a whole lot about the relation between gender and diabetes risk. All you can say is that there's no strong evidence for a gender difference in risk.

Lastly, there’s BMI. **The odds ratio for a unit increase in BMI is exp(0.073879) = 1.08, with 95% CI 1.03 to 1.13, *p*=0.00153**(or 0.002 to three decimal places, as is usual reporting practice). That’s a low *p*-value, so you can conclude that people with higher BMIs are more at risk of diabetes. That’s entirely what we'd expect given that 90% of diabetes cases are type 2, which is associated with lifestyle, which affects one’s BMI.

**Assessing Model Fit**

Welcome to the final week of the course! In this week, you will learn how to assess model fit and model performance, how to avoid the problem of overfitting, and how to choose what variables from your data set should go into your multiple regression model. You will put all the skills you have learned throughout the course into practice. By the end of this week, you will be able to evaluate the model assumptions for multiple logistic regression in R and describe and compare some common ways to choose a multiple regression model.

Objectives:

* Evaluate the model assumptions for multiple logistic regression in R
* Describe and compare some common ways to choose a multiple regression model

**Model fit in logistic regression**

If a regression model is to be useful, it needs to fit the data well. What does that mean?

**There are essentially two very different ways of approaching this question: predictive power and goodness of fit.** Ideally, you want your model to do well on both. Some of what I'll cover is also relevant to other types of regression, and the course on linear regression in this series introduced the important concept of the **residual**.

With the first, **external validation**, the aim is to get a statistic that measures how well you can predict the dependent variable (the outcome, so getting diagnosed with diabetes in our case) based on the independent variables (the predictors, such as age and BMI). This will tell you about the “predictive power” or “explanatory power” of the model. Generally, they range between 0 (the model explains none of the data and the variables don’t predict the outcome at all) and 1 (the variables predict the outcome completely). These include measures such as R-squared and the area under the ROC curve, which we’ll look at shortly.

**The other approach to evaluating model fit is to compute what’s called a goodness-of-fit statistic.**These kinds of **internal validation** measures include the deviance and the popular Hosmer-Lemeshow statistic. There are formal tests for these that yield a *p*-value, so if you’re happy to use the usual cut-off of *p*=0.05, you can use them to decide whether your model fits the data acceptably. Again, more on those shortly. **Goodness of fit tells you nothing about predictive power – and vice versa**. You can get good prediction with poor fit or a model with good fit but poor prediction.

Let’s look at some of these in brief, beginning with what they are and then how to calculate them in R.

**R-squared measures**

The previous course on linear regression models covered how to test how well your linear regression model fits the data. The most common way to do so is with the R-squared value, which measures the proportion of the variance in the outcome variable (Y) that can be explained by your predictor variables (X1, X2… etc). **An R-squared value close to 1 indicates strong predictive power, while one close to 0 indicates poor predictive power**. As you now know, logistic regression is used when the outcome variable is binary. We can't do correlation tests if your Y can only take 2 values – so what can we do?

It turns out that there are many ways to approximate an R-squared for logistic regression. One of the best ways is with the McFadden (pseudo) R-squared. This measure depends on the “likelihood” of your model, which is a cryptic way of describing how compatible your model parameters are with the observed data. You don’t need to know the details of the calculation of the likelihood or indeed of McFadden’s R-squared measure. The McFadden R-squared can be interpreted in a similar way to the “classic” R-squared from a linear regression: high values are best. In practice, though, R-squared values – whether the McFadden version or any other – tend to be low and certainly lower than people who are used to linear models expect. This does not mean the model is bad – it's more a reflection of the limitation of the R-squared measure than of the model.

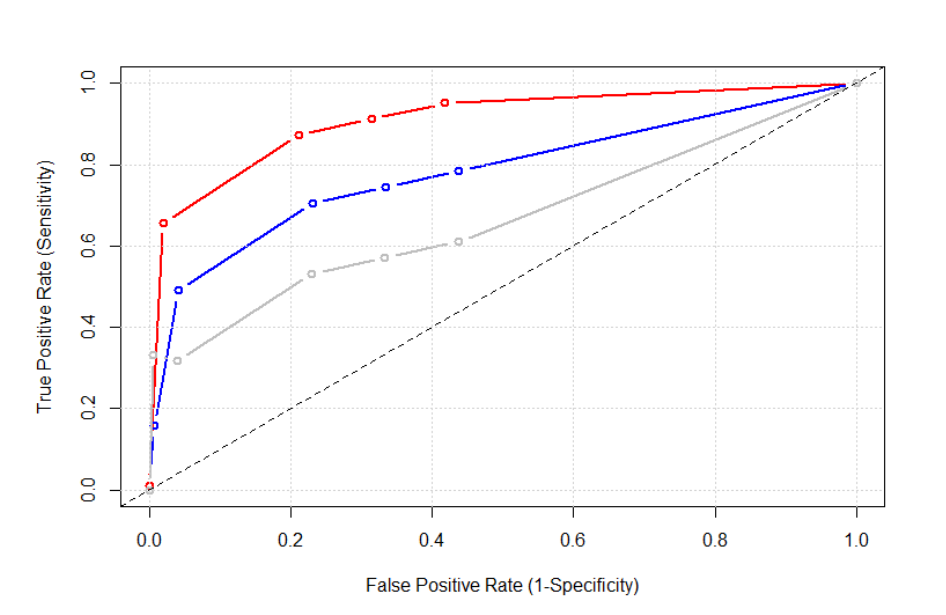
**Discrimination: c-statistic or area under the ROC curve**

**In prognostic modelling – estimating the risk of an outcome (e.g. disease) based on a person’s characteristics (e.g. age, gender, etc) – we want to be able to assess a model’s discrimination**. Discrimination is a “measure of how well the model can separate those who do and do not have the [outcome] of interest” [Nancy Cook in Circulation 2007]. In our case, we’re interested in distinguishing between people with and without a disease (diabetes). When looking at a sample of patients, a model with good discrimination will declare those with a disease to have had, on average, a higher risk than all of those without the disease.**Therefore, in the modelling world, discrimination is a good thing.**

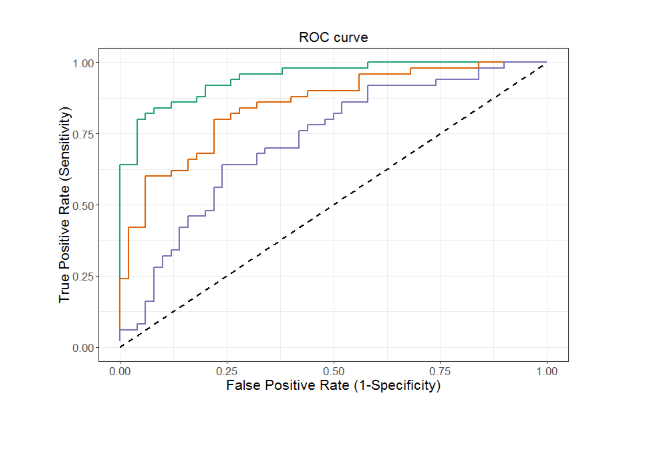
You can see why you would want to test this for your logistic models. In the example you’ve done, you've built a model to test the potential relationship between several traits (age, gender, cholesterol level, etc.) with a disease outcome: diabetes. **In your sample of patients, would your model predict a higher risk score for those who we have observed to have diabetes than those who don’t?**

One of the most popular ways to do this is called the “area under the receiver operating characteristic (ROC) curve, or “c-statistic” for short. The ROC is a plot of sensitivity (probability of a positive result among the cases) against 1 - specificity (probability of a negative result among the non-cases). A “case” here is someone with the disease or outcome of interest. This can be reworded as the “true positive rate” vs. the “false positive rate”.

These terms are covered in more detail in the epidemiology course as part of the online MPH. This is what the plot looks like for 3 models on dummy data:



**The area under a curve, which is calculated by a technique called integration, is the c-statistic.** A c-statistic of 0.5 indicates that the model is only as good at predicting the outcome as random chance (i.e. no discrimination). A curve at or close to the black line (y = x) in the diagram would be an example of this. **As the curve pulls away from the black line, the area under it increases, so therefore the discrimination increases**. In the diagram, the model represented by the red ROC curve has the best discrimination. A c-statistic of 1 would be perfect, but of course this never happens in real life and in fact, as Cook’s article shows, the theoretical maximum for a given model is often lower than this. A c-statistic below 0.5 would predict the outcome worse than random chance, which would mean a very poor model indeed. With large datasets, these plots will be smooth, but for smaller ones they are often somewhat jagged like this:



The c-statistic appears a lot in the machine learning literature too when the aim is prediction, as it is here.

**Deviance**

In regression it concerns how well – or rather how badly – the model fits the data. **It’s a measure of the “goodness of fit”**. In a linear model, where the outcome can take on any value, the predicted value can match the actual outcome exactly or differ from it by a measurable amount. This leads straightforwardly to the concept of **deviance – a measure of how the prediction differs from the observed outcome**. In logistic regression, however, the observed outcome can only take on two values, zero and one, whereas the predicted value, a log odds, can take on any value and can be mapped to a probability, which can take on any value between zero and one.**Therefore, we can’t just take the deviance measure from the linear regression case. Some adjustment is necessary.**

One very common approach can be taken when the data can be aggregated or grouped into unique “profiles”: groups of cases that have the same values of the predictors, e.g. patients with the same age, gender and insurance. After fitting the model, we can get an observed number of events and an expected number of events for each profile. **The two well-known statistics for comparing the observed number with the expected number are the deviance and Pearson’s chi-squared**. Both produce statistics that can be compared against tables of a chi-squared distribution in order to see how unusual the value of the statistic is for that model, which yields a *p*-value. High *p*-values (e.g. above the usual threshold of 0.05) mean that the model’s deviance statistic is nothing unusual, which is a good thing as it means that the model fits the data well. **The deviance compares our model with a selected number of candidate variables in it against one that fits the data perfectly – the “saturated model” – to see whether we’re missing anything important, such as interactions between variables or non-linearities**.

This profile approach is fine when we have only categorical predictors. It will likely be fine with age (in years) if there are several patients each with a different value for age. If the data are spread so that there’s only one case per profile, then the deviance and the Pearson’s chi-squared statistic won’t fit the chi-squared distribution very well at all, so the test breaks down. What can we do? This leads us to the next measure: the Hosmer-Lemeshow statistic, proposed in 1980 and still very much in use today.

**Calibration: Hosmer-Lemeshow statistic and test**

Here, patients are grouped together according to their predicted values from the model, which are ordered from lowest to highest and then separated into typically ten groups of roughly equal size. For each group, we calculate the observed number of events – here that’s the number of patients with diabetes – and the expected number of events, which is just the sum of the predicted probabilities for all the patients in the group. Pearson’s chi-squared test is then applied to compare observed counts with expected counts. **A large *p-*value (e.g. above the conventional 0.05) indicates that the model’s predicted values are a good match for the real (observed) values, i.e. the model is a good fit**.

**The authors’ own work has revealed some limitations with the test.**With small data sets, the test has limited ability (limited power) to detect important differences between the observed and expected counts i.e. to detect poor fit that’s poor enough to worry about. At the other end, with large data sets, you can get a low *p*-value when the difference between the observed and expected isn’t that important.

It has some other issues too. One is that although the standard number of groups to use is ten, the *p*-value can be altered just by choosing fewer or more than ten groups, and there’s no good way of deciding on the number. Another is that some people, including me, have found that adding interaction terms between variables, even non-significant ones, can alter the statistic.

Despite all of these problems, the test is much used and reported in the literature. These days, I only use the plot of the observed against the expected rather than the *p*-value. Some other tests are reviewed at <https://support.sas.com/resources/papers/proceedings14/1485-2014.pdf> though the article uses SAS as the exemplar software package.

**How to get these statistics in R**

The deviance is given by default but the R-squared, c-statistic and Hosmer-Lemeshow statistics and test must be requested in R.

## How to Interpret Model Fit and Performance Information in R

Let’s consider the last model we ran, which had age, cholesterol and insurance. How well does this model fit the data? We’ll start with what R gives us by default and decide what’s useful and what’s missing. R’s default output for the “glm” command includes the following:

* The call to the algorithm itself, i.e. what the model is
* Deviance residuals
* Coefficients and their standard errors
* Dispersion parameter
* Null deviance
* Residual deviance
* AIC
* Number of Fisher scoring iterations

The number of Fisher scoring iterations presented means that the model has converged, i.e. the algorithm has worked and found the best solution. In the others, there’s quite a bit of mention here about “deviance”, which was described in the earlier reading. **The null deviance tells us about model fit with just the intercept term in it**. What’s more important is the deviance when we’ve added our predictors and how much the deviance falls when compared with the null value. For our model:

* The null model had a deviance of **334.54 on 388 degrees of freedom**
* Our model had a residual deviance of **289.28 on 384 degrees of freedom**

That’s a difference of 334.54-289.28 = 45.26 at a “cost” of 388-384 = 4 degrees of freedom. If you recall the concept of degrees of freedom from the previous course, then you’ll see that the four d.f. represent four added parameters, which came from one for age, one for cholesterol and two for insurance. **Our model has improved by 45.26 for our “investment” of 4 d.f., but is that a good return on our investment?**

To answer this, we need to define “deviance” in this context. This can get very technical, as there are different ways to compute this. First, the bigger the deviance, the worse the model fits the data, so you want to be able to test this. Second, we want our model to be an improvement on the null model – if you have at least one variable with a low *p*-value, then you’ll have an improvement.

**Null Deviance and Residual Deviance**

**To understand residual deviance, we must first think about 3 models: the null model, the proposed model and the saturated model**.The null model, as discussed earlier, is one where we only include the intercept: this model therefore only has one parameter. The proposed model is the model with the variables we included in our logistic regression. The number of parameters in the proposed model is the number of variables plus one (the intercept): this is because each of the variables we have included only needs one parameter, but remember that a categorical variable with three categories, for example, will need two parameters. The saturated model is a model which fits the data perfectly, because it has as many parameters as there are data points.

**The null deviance is a measure of how well the null model explains the data compared with the saturated model.**Having just one parameter, the null model does not usually explain the data very well, and this is indicated by a large null deviance. The point of doing a regression model is that we reckon we can do better at explaining the data with a few variables (age, sex, etc). This brings us on to the residual deviance: how well the proposed model explains the data compared with the saturated model.

**The difference between the null deviance and the residual deviance gives us an idea of how well our model has performed (at the cost of degrees of freedom)**. It’s like a return on investment: how much benefit (explanation of the variation in the outcome) do we get for our investment (the variables we’ve added or, more accurately, the degrees of freedom taken up by those variables). **If the model is “good”, then the difference between the null deviance and the residual deviance will be large**. There are formal ways to see whether the difference is large enough.

**To test whether each added parameter decreases the deviance by a significant amount, we ask R to compare it with a chi-squared value for the number of degrees of freedom lost.** If the *p*-value is low, it indicates that the corresponding added variable causes a significant change in deviance, and thus leads to a better fitting model. It’s not at all essential that you understand why we use the chi-squared distribution for this comparison – just that you know how to interpret the resulting *p*-value.

In our case, adding the variables age and cholesterol significantly reduce the deviance and improve the model fit, as indicated by their low *p*-values, but including the insurance variable does not improve the model fit enough to justify the loss in degrees of freedom, as indicated by its high *p*-value of 0.2896.

**AIC**

Lastly, I’ll mention the AIC. **This is short for Akaike Information Criterion and measures the quality of a model in terms of the amount of information lost by that model**. It therefore recognises that all models lose information compared with “reality”, but some models lose less than others. It’s of no use by itself but is used for comparing two or more models. The lower AIC value represents the best model.

**Ways to choose your model**

As I explained in the videos, the problem is that having too few predictors leads to poor prediction, but having too many can cause overfitting, non-convergence, difficulty in interpretation and explaining results to other people. **So how do you choose the predictors?**

It’s always a good idea to begin by reviewing the relevant literature and expert knowledge, but that will only get you so far. It may tell you that you should include age, gender and perhaps a few other things, but it’s very likely that you’ll have other variables in your data set that are worth trying. One option is to enter and keep them all in your model, whatever the *p*-values. This is a good idea if you don’t have too many and/or you can use *a priori* knowledge for them all.

**A variant of this is backwards elimination, where you then drop the non-significant variables.** This works okay in some circumstances, but you need to check for correlation between variables. The best way to do that is by inspecting the odds ratios for the predictors that you are keeping – first with all variables in the model and second when you drop some. If the odds ratios change noticeably when you drop some, then you’ll need to add back at least one of the dropped ones.

Forward selection involves starting with an empty model and trying variables one at a time. Stepwise selection involves a mixture of forwards and backwards. **Both should be avoided**. Similarly, “all-possible-regressions", which literally tests all possible subsets of the set of potential independent variables, is to be avoided. It might sound like it, but it’s not in fact guaranteed to give you the best model for your data.

**Training and testing data sets**

Where possible, it’s good practice to split a data set into training and testing components. Then apply the training set model to the test set data. If you get very different answers, then rethink the complexity of the training set model and repeat. This is standard practice in machine learning because of the risk of overfitting and overtraining, in which the algorithm fits the data too closely so it's unable to distinguish between signal and noise in the data set and therefore performs badly on a new data set. **However, it's also strongly recommended with statistical models where possible**. Another related technique is called *k*-fold cross-validation, which is often used when you have limited data.

With small data sets, splitting into training and testing sets is not advisable. The data sets we are using in these statistics for public health courses are considered small, which is why I haven’t suggested splitting them.

The majority of what you included was not significantly associated with the outcome. There’s potential for “**clearing out the garbage**” by removing the non-significant variables, **but you would need to check that this doesn’t alter the odds ratios for the variables that remain in the model**. Also, with only 11 variables, the table of results isn’t too big – and it’s important to know which variables were not associated with the outcome in this data set. **So-called "negative findings" are just as important to know about and report as "positive findings"**.

Another reason for reporting all the variables is that sometimes the *p*-value will be very close to the conventional 0.05 threshold – a bigger sample size could have made it smaller, which just emphasises the **arbitrary nature of this threshold**and why there’s something of a backlash against the “tyranny of the *p*-value” in the statistical literature.

**Optional further reading on model selection methods**

There’s more detail on how stepwise selection is done and why it’s a bad idea at <https://people.duke.edu/~rnau/regstep.htm> You can’t trust the coefficients and you can’t trust the *p*-values. **That’s true for all types of regression**, including Cox that we’re covering in the next course in this specialisation.

**Rather than use *p*-values for each variable to decide whether to include it, a (better) alternative is to be guided by a quantity that I mentioned earlier called the AIC**, Akaike’s Information Criterion, or a similar one called the Bayes Information Criterion (BIC). Without going into the maths behind the “information” (which has a technical meaning here), you just need to know that **the AIC aims to describe how well the model fits the data** while penalising models with lots of coefficients and that **lower values of the AIC are desirable**. It's useful for comparing models, but if you only run one model it's of no value.

Some people use the R-squared statistic to pick the “best” model. **The R-squared estimates the predictive power of the model and tells you nothing about how well the model fits the data** (it is an external validation technique). It’s not ideal but it does give you useful information. With logistic regression, the maths behind the R-squared is trickier than for linear regression, leading to many different versions of it being proposed. **I'd advise against using only the R-squared to pick your best model**.

**An alternative to choosing a single model is to use model averaging**. This has a different philosophy from the model selection methods by considering that there are several “good” candidate models and that we don’t have to choose between them – **we can just take an average**. This idea has been developed and applied to all kinds of regression. This has been quite a large area of methodological research for some time, and model averaging can be done in R with functions written by users. **It’s now often used to average regression coefficients across multiple models with the goal of capturing a variable's overall effect**. This use of model averaging implicitly assumes the same parameter exists across models so taking an average is a sensible thing to do. At first glance, this assumption seems reasonable, but regression coefficients associated with variables might not have the same interpretations across all of the models in which they appear, and that makes interpreting the averaged value tricky. Despite the issues – after all, there are issues with every method in existence – model averaging is widely used, but due to the size of the literature and the technical details of **Bayesian model averaging** I’ll go no further. A readable summary of the area is given by <https://warwick.ac.uk/fac/sci/statistics/crism/research/17-06/17-06w.pdf>. The focus is on Bayesian model averaging – **in Bayesian statistics, one combines the data with one’s prior beliefs to produce the model output, whereas in classical or “frequentist” statistics, one is driven only by the data**. The article also covers the frequentist approach.