



Back to basics: What do we do?

We create *measurements* of psychological traits, like "intelligence", "personality", "values" etc.

In practice, this means *assigning numbers* to unobserved, latent qualities of individuals.

In Classical Test Theory, we get the numbers from calculating the sum of scores or ratings from a test = "raw score". Then we use statistical methods to interpret the raw score.

In Item Response Theory, we get the numbers from applying statistical models to each item in a test.

Classical Test Theory (CTT)

Assumed model:

$$\text{Observed Score} = \text{True Score} + \text{Error}$$

CTT deals with sums of scores on entire tests or subtests. The reliability of the test is a central concept in CTT.

The goal is to estimate the examinee's "true score" for the test, and the uncertainty (error) in this estimation.

Error and reliability are estimated from the variability of a sample of observed scores.

CTT is based on frequentist statistics, where repeated sampling and the central limit theorem are important concepts.

Classical Test Theory (CTT)

Pros

- Easy to implement – the calculations are very simple
- Proved as useful in the realm of fixed form psychological tests

Cons

- The idea of a true score makes little theoretical sense. Does every possible test measure a distinct psychological trait?
- Error is assumed to be equal for all examinees – which isn't true
- Items are treated as fixed parts of the test, and cannot be treated separately.

This makes CTT unsuitable for item banking or adaptive testing, where different sets of items are presented to the examinees.

Item Response Theory (IRT)

Assumed model (2 parameters):

$$\Pr(x_i = 1 \mid \theta, a_i, b_i) = \frac{1}{1 + e^{-a_i(\theta - b_i)}}$$

"The probability that the response x on item i is correct, *given* the true trait of the examinee, θ , and the item parameters a_i and b_i ."

The model shows the relationship between the item parameters and the examinee trait.

So, if we observe an individual's response to an item, and we know the item parameters, we can use this model to estimate the trait of the individual θ .

Item Response Theory (IRT)

Pros

- Items are independent – can be put together in flexible ways
- Allows for Computerized Adaptive Testing (CAT)
- Higher precision of scores compared to CTT
- Theoretical advantages – traits are modelled directly, item properties and estimation error are allowed to vary, etc.

Cons

- Harder to implement – more computation needed for scoring
- Harder to understand – the statistical model, estimation algorithms etc. are quite complex

We do a lot of stats!

- Recap
 - What is probability?
 - What is a probability distribution?
 - What is a statistical model?
- Statistical inference
 - Estimating correlation coefficients
 - "Fitting" linear regression models
- Significance testing
 - t-test (one sample and two samples)
 - Anova/F-test
 - Problems with null hypothesis significance testing
- Statistical prediction



Probability theory

What is probability?

Frequentist interpretation:

The *relative frequency of occurrence* of an experiment's outcome, when repeating the experiment.

$$\Pr[E] \equiv \lim_{n \rightarrow \infty} \frac{n_{E \in E}}{n}$$

Bayesian interpretation:

Our *state of knowledge or state of belief* in a hypothesis, given prior knowledge and empirical data.

$$\Pr[H|D] = \frac{\Pr[D|H] \Pr[H]}{\Pr[D]}$$

Pearson 156

Probability distributions

What is a probability distribution?

- A norm group is an example of a probability distribution
- Empirical = observed sample values (e.g. total score distribution)
- Theoretical = expected population values (e.g. ability distribution)

Pearson 157

Gaussian/Normal distribution functions in R

- Normal density function (x: quantile/z-score)
`dnorm(x = 0, mean = 0, sd = 1)`
`curve(dnorm, from = -3, to = 3)`
- Normal distribution function (q: quantile/z-score)
`pnorm(q = 0, mean = 0, sd = 1)`
`curve(pnorm, from = -3, to = 3)`
- Quantile function (p: cumulative probability)
`qnorm(p = 0.5, mean = 0, sd = 1)`
`curve(qnorm, from = 0, to = 1)`
- Random sampling from a standard normal distribution (n: sample size)
`x <- rnorm(n = 100, mean = 0, sd = 1)`
`hist(x)`

Statistical models

What is a statistical model?

- A theoretical representation of our assumptions of the data generation process "in the real world", involving randomness.
- Random variables = not fully observed/not deterministic. Often assumed to follow a normal distribution (central limit theorem):

$$X \sim N(\mu, \sigma^2)$$

- All models are wrong! (George Box)

Statistical inference

Statistical inference

What is statistical inference?

- Assume a statistical model, e.g. $X \sim N(\mu, \sigma^2)$
- Estimate parameters, e.g. $\hat{\mu}$ and $\hat{\sigma}^2$, from a sample of observations:

$$\hat{\mu} = \bar{x} = \frac{1}{n} \sum_{i=1}^n x_i, \quad \hat{\sigma}^2 = s^2 = \frac{1}{n-1} \sum_{i=1}^n (x_i - \bar{x})^2$$

- Estimate the error of the estimated parameter, e.g:

$$SE_{\bar{x}} = \frac{s}{\sqrt{n}}$$

- The confidence interval of the parameter can be calculated using the standard error, e.g:

$$95\% \text{ CI} = \bar{x} \pm 1.96 \times SE_{\bar{x}}$$

Statistical inference

Common calculations for standard parametric models:

- Arithmetic mean \bar{x} : `mean(x)`
- Sample variance s^2 : `var(x)`
- Sample standard deviation s : `sd(x)`
- Standard error of the mean $SE_{\bar{x}}$: `sem <- sd(x) / sqrt(n)`
- 95 % Confidence Interval: `mean(x) - 1.96 * sem`
`mean(x) + 1.96 * sem`

Other useful statistics:

- Sample median: `median(x)`
- Sample quantiles: `quantile(x)`
- Sample max value: `max(x)`
- Sample min value: `min(x)`

Correlation

Goal: Estimating the strength of a (linear) relationship between two (paired) random variables.

Assume that we know the parameters of two random variables:

$$X \sim N(\mu_X, \sigma_X^2), \quad Y \sim N(\mu_Y, \sigma_Y^2)$$

Then the correlation parameter in the population is:

$$\rho_{XY} = \frac{\text{cov}(X, Y)}{\sigma_X \sigma_Y} = \frac{E[(X - \mu_X)(Y - \mu_Y)]}{\sigma_X \sigma_Y}$$

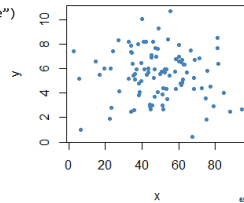
Which can be estimated from the sample correlation coefficient:

$$\hat{\rho}_{XY} = r_{xy} = \frac{\sum_{i=1}^n (x_i - \bar{x})(y_i - \bar{y})}{n s_x s_y}$$

Correlation

Let's calculate a correlation coefficient for simulated data:

```
x <- rnorm(100, mean=50, sd=20)
y <- rnorm(100, mean=5.5, sd=2)
cor.test(x, y, method = "pearson")
plot(x, y, pch = 20, col = "steelblue")
```



Pearson

65

Correlation: output

```
> cor.test(x, y, method = "pearson")

Pearson's product-moment correlation

data: x and y
t = -0.77869, df = 98, p-value = 0.438
alternative hypothesis: true correlation is not equal to 0
95 percent confidence interval:
 -0.2786669  0.1198463
sample estimates:
 cor
-0.07841779
```

Estimated
correlation
coefficient

95%
Confidence
Interval

Pearson

66

Simple regression

Goal: Estimating two coefficients α (intercept) and β (slope) for the linear function:

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

Assume that the model errors are normally distributed with mean 0:

$$\varepsilon \sim N(0, \sigma_\varepsilon^2)$$

Then the least-squares estimators $\hat{\alpha}$ and $\hat{\beta}$ are unbiased with regards to the "true" coefficients:

$$\hat{\beta} = r_{xy} \frac{s_y}{s_x}, \quad \hat{\alpha} = \bar{y} - \hat{\beta} \bar{x}$$

Pearson

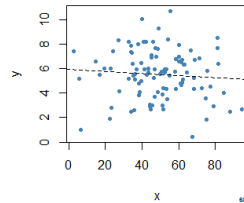
67

Simple regression

Let's fit a regression model to our simulated data. This will use ordinary least squares (OLS) to find the best fitting line for the data (which minimizes the sum of squared errors for all observations):

```
reg1 <- lm(y ~ x)
summary(reg1)
confint(reg1)

abline(reg1, col = "black", lty = 2)
```



Simple regression: output

```
> summary(reg1)
```

```
Call:
lm(formula = y ~ x)
```

```
Residuals:
    Min       1Q   Median       3Q      Max
-3.5416 -1.2879 -0.2987  1.0342  4.3313
```

```
Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)  5.577362    0.502164   11.107  <2e-16 ***
x           -0.007029    0.009026   -0.779    0.438
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
Residual standard error: 1.767 on 98 degrees of freedom
Multiple R-squared:  0.006149, Adjusted R-squared:  -0.003992
F-statistic: 0.6964 on 1 and 98 DF, p-value: 0.438
```

Estimated coefficients for intercept $\hat{\alpha}$ and slope of predictor x $\hat{\beta}$

Standard error of the estimates. Use `confint(reg1)` to get 95% Confidence Intervals

What's with the '~'?

When defining statistical models in R, you can often use "formulas". These are the relationships between dependent variables (DV) and independent variables (IV) that we want to model:

DV ~ IV

There can be more than one IV, as in the case of multiple regression, which makes this syntax very flexible:

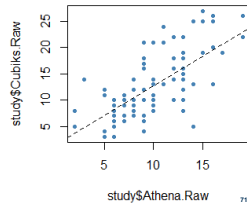
DV ~ IV1 + IV2 + IV3

For simple regression, both the DV and the IV are continuous variables, but the `lm` function can be used with categorical IVs as well.

Exercise: Athena study

Do the same thing with the data from the Athena validity study!

1. Use the dataframe study from previous slides
2. Calculate correlation between Athena.Raw & Cubiks.Raw
3. Make a scatterplot
4. Fit a linear model and save it as reg2
5. Add the line from the model to the plot



Pearson

Significance testing

Significance testing

Goal: calculating the probability of observed data given an assumed NULL model, based on a theoretical (or bootstrapped) sampling distribution.

E.g. for a one-sample t-test of a sample mean:

$$t = \frac{\bar{x} - \mu_0}{SE_{\bar{x}}}, \quad df = n - k$$

The t-statistic is "the difference between the observed mean and the expected mean (which is 0) divided by the standard error of the mean".

Note the importance of the standard error in the formula! What happens if n becomes infinitely large?

Pearson

173

Significance testing

Assuming that the t-statistic follows a known distribution (Student's t), the probability of the observed t-value can be looked up in a t-table, or by using the Student's t distribution function.

If the p-value is below a threshold (commonly .05), the NULL model is rejected and it is concluded, in this example, that the sample mean is "significantly different from 0".

The t-test is commonly used for significance testing of other parameters as well – like correlation coefficients and regression weights.

Simple regression

For our Athena regression model, we can calculate the t-statistic for the slope coefficient:

$$t = \frac{\hat{\beta} - \beta_0}{SE_{\hat{\beta}}} \approx \frac{1.1308 - 0}{0.1294} \approx 8.739, \quad df = n - k = 89 - 2 = 87$$

Then we can calculate the p-value by using the distribution function of Student's t-distribution. If we're doing a two-tailed test:

```
> 2 * pt(q = 8.739, df = 87, lower.tail = FALSE)
[1] 1.537233e-13
```

This is very small, and close to the values from our regression model. The difference is caused by rounding when calculating the t-value by hand.

Simple regression

```
> summary(reg2)
```

Call:
lm(formula = Cubiks.Raw ~ Athena.Raw, data = study)

Residuals:

Min	1Q	Median	3Q	Max
-12.2274	-3.3121	-0.4429	3.0341	10.1648

Coefficients:

	Estimate	Std. Error	t value	Pr(> t)
(Intercept)	1.3968	1.3931	1.003	0.319
Athena.Raw	1.1308	0.1294	8.735	1.56e-13 ***

Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 4.497 on 87 degrees of freedom
Multiple R-squared: 0.4673, Adjusted R-squared: 0.4611
F-statistic: 76.31 on 1 and 87 DF, p-value: 1.563e-13

Inference:
Estimated
parameters and
standard errors for
 $\hat{\alpha}$ and $\hat{\beta}$

Significance tests:
t- and p-values of
the observed
estimates $\hat{\alpha}$ and $\hat{\beta}$

Two groups t-test

Let's do t-tests on simulated data. I won't go into details about the formulas this time :)

- First, create grouping variable
(2 levels w 50 cases each): `g <- gl(n = 2, k = 50)`
- t-test (independent groups): `t.test(formula = x ~ g)`
- Plot the model: `boxplot(x ~ g)`

Why do we use the formula syntax `x ~ g` for t-tests?

Two groups t-test

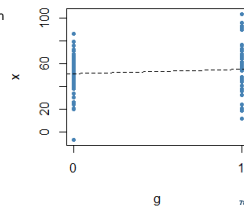
T-tests are actually based on a special form of OLS linear regression!

The group means are statistics that minimize the sum of squares in this model = least squares estimation.

You can think of this as drawing a line through the group means, as in the plot to the right.

Try this: `summary(lm(x ~ g))`

The intercept in this model is the mean of group 1, and the slope coefficient is the difference between the means of the two groups.



Two groups t-test

```
> t.test(formula = x ~ g)
```

Welch Two Sample t-test

data: x by g

`t = -0.99967, df = 95.957, p-value = 0.32`

alternative hypothesis: true difference in means is not equal to 0

95 percent confidence interval:

`-11.408137 3.766172`

sample estimates:

mean in group 1 mean in group 2
51.07788 54.89878

t- and p-values for the observed mean difference

95% Confidence Interval for the mean difference

Analysis of variance

Analysis of variance (Anova) is commonly used to detect mean differences between groups. Let's do a one-way Anova on simulated data!

- Create new grouping variable
(4 levels w 25 cases each): `g <- gl(n = 4, k = 25)`
- Anova function: `aov1 <- aov(formula = x ~ g)`
- Anova table: `summary(aov1)`
- Plot the model: `boxplot(x ~ g)`
- Group means: `by(x, g, mean)`
- Model coefficients: `summary.lm(aov1)`

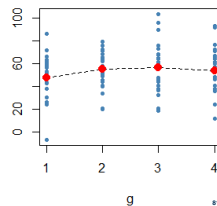
Analysis of variance

Surprise! The Anova is also based on a OLS linear regression, just like the t-test!!

Instead of modelling the difference between two means, the coefficients are now the differences between the group means and the mean of the baseline group (group 1).

This is how the comparisons work:
`contrasts(aov1)`

We rarely report the coefficients of this model. \times
Instead, we're more interested in the F-statistic which is a summary of the entire model.



Analysis of variance: output

```
> summary.lm(aov1)
```

```
Call:
aov(formula = x ~ g)
```

```
Residuals:
    Min       1Q   Median       3Q      Max
-55.885 -11.275  -0.516  16.314  41.202
```

```
Coefficients:
              Estimate Std. Error t value Pr(>|t|)
(Intercept)   54.596     3.903   13.989  <2e-16 ***
g2             -8.986     5.519   -1.628    0.107
g3             -3.243     5.519   -0.588    0.558
g4              2.150     5.519    0.390    0.698
```

```
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
Residual standard error: 19.51 on 96 degrees of freedom
Multiple R-squared:  0.046,    Adjusted R-squared:  0.01618
F-statistic: 1.543 on 3 and 96 DF,  p-value: 0.2084
```

Results from the Anova: F-test with p-value

Same as in the summary of a regression model!
Useful information about model coefficients (= group mean differences)

Analysis of variance

What is the F-statistic?

The Anova is actually a *comparison of models*:

1. Two linear models are fitted on the data - the null model (grand mean) and the model including the grouping variable(s)
2. The "model fit" of these models are compared by partitioning the Sums of Squares into explained and unexplained variance
3. A significance test is calculated using the F-distribution

Analysis of variance

The results from the F-test is what we commonly report from conducting an Anova: $F(3, 96) = 1.543$; $p = 0.208$.

This tells us the probability of our observed group differences, given the null model; that all observations are sampled from the same population.

Again, if $p < .05$ we reject the null model. This is an indication that the group means are different somehow.

To follow up a F-test, we can for example use Tukey's HSD for pairwise comparisons of the group means:
`TukeyHSD(aov1)`

Some problems with significance testing

1. p-values are hard to interpret. A common misconception is that p is the probability that the null hypothesis is true. In fact, the p-value never gives support for any statistical model!
2. p-values only tell us is only the probability of our observations given the null model. Is this really what we want to know?
3. The null hypothesis is always wrong. It's unreasonable to think that any parameter is exactly 0 in the population.
4. The use of $p < .05$ as a cut-off for "statistical significance" is arbitrary at best. With a large enough sample, any finding can pass this threshold.

"What's wrong with [null hypothesis significance testing]? Well, among many other things, it does not tell us what we want to know, and we so much want to know what we want to know that, out of desperation, we nevertheless believe that it does!"

Cohen, 1994



What should we do instead?

Cohen (1994) suggests the following:

1. Exploratory data analysis using graphic methods
2. Improvement and standardization of measurements
3. Greater emphasis on estimating effect size
4. Estimate confidence intervals
5. Informed use of available statistical models



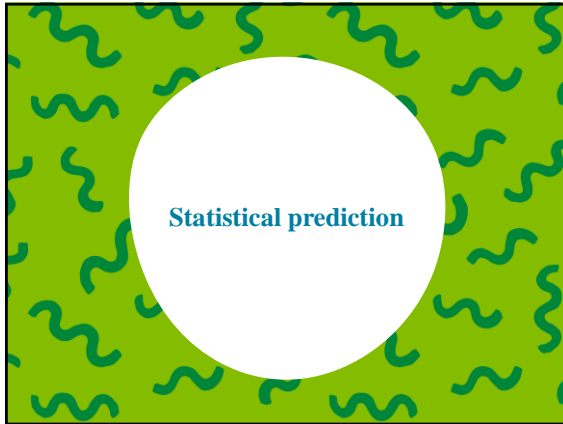
187

How do we do this in R?

1. Always plot your data using `hist()`, `boxplot()` and `plot()`.
2. This is already our job...
3. Keep reporting correlation coefficients, which is a measure of effect size. Also calculate Cohen's d when studying group mean differences.
4. Calculate and report confidence intervals using the `confint()` function.
5. This is what we're accomplishing with this course ☺



188



Statistical prediction

Prediction is when we use a statistical model, with parameters estimated from observed cases, to predict the most probable values of unobserved cases.

For example, if we know that the raw score on Athena for an individual was 20, we can predict their most likely raw score on Cubiks. Remember the linear regression model:

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

We estimated $\hat{\alpha} = 1.3968$ and $\hat{\beta} = 1.1308$, and since the average (or expected) error $E[\varepsilon] = 0$, our prediction becomes:

$$\hat{y}_i = \hat{\alpha} + \hat{\beta} x_i + 0 = 1.3968 + 1.1308 \times 20 = 24.0128$$

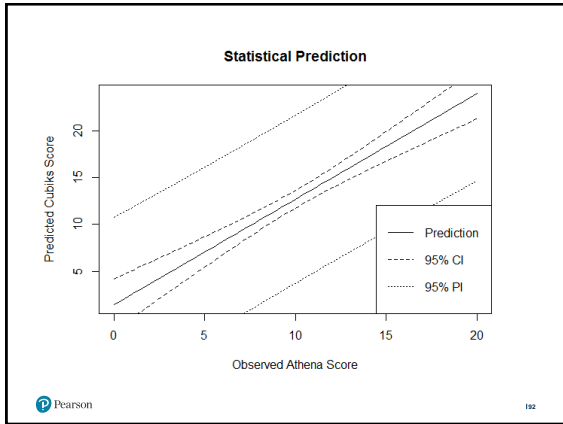
Statistical prediction

Prediction is easy with R – just create a data.frame with the observations you want to predict and add it to the predict function:

```
> new_data <- data.frame(Athena.Raw = 20)
> predict(reg2, newdata = new_data)
      1
24.01199
```

All predictions have a degree of uncertainty. You can get a 95 % prediction interval (note: different from a *confidence* interval) like this:

```
> predict(reg2, newdata = new_data, interval = "prediction")
      fit      lwr      upr
1 24.01199 14.67072 33.35326
```



Statistical learning & AI

Statistical Learning and Machine Learning are similar things, coming from two different academic fields: Statistics and Computer Science.

The goal is to create statistical models or algorithms that learn from experience in an automatic way.

These models are the basis of modern developments in Artificial Intelligence (AI).

The basic ideas are pretty simple and similar to what we do – combining inference (estimating/updating our understanding of the world) with prediction (using our understanding to anticipate the future). Finally, actions are based on the predictions, and the process is repeated.

Pearson 193

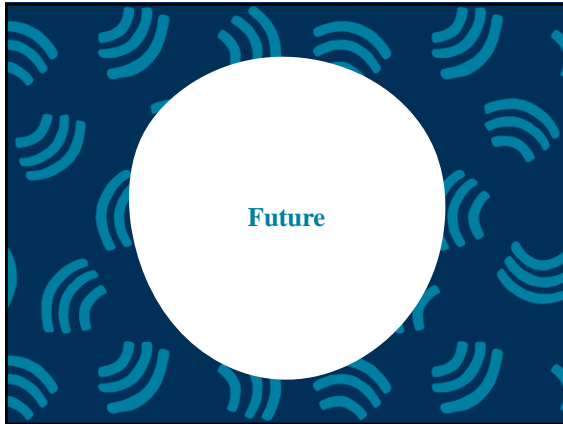
Statistical learning & AI

Are we making best use of our data?

What kind of predictions do we want to be able to make?

Can we apply a statistical learning approach in our work?

Pearson 194





There is so much more!

More things you can do in R:

- Generalized Linear Models – e.g. logistic regression, poisson regression etc.
- Non-parametric statistics – e.g. the bootstrap
- Factor analysis – e.g. PCA
- Structural Equation Modeling
- Bayesian analysis
- Monte Carlo studies
- Cross-validation
- Tree-based methods
- Neural networks
- Etc...

Once you know the standard approaches, learning new stuff is not that hard!

 Pearson



GitHub Repository

This is where I'm going to share code and functions that are useful for our day-to-day work.


To install or update the "talentlens" package:

```
devtools::install_github("talentlens/talentlens")
```

I plan to put the following functions in there:

- Getting data from Concerto
- Test scoring
- IRT theta estimation

Etc.

 Pearson 197

ALWAYS LEARNING

Bonus: the bootstrap

The bootstrap is non-parametric, which means that it doesn't rely on any theoretical distribution.

This makes it very flexible – the bootstrap can be used to approximate the sampling distribution of ANY statistic!

Instead of assuming that observed data was generated from a normal probability distribution, for example, the bootstrap is assuming that the observed data has its own probability distribution.

By sampling from observed data repeatedly, with replacement, and calculating the statistic of interest each time, we get an approximate sampling distribution of the statistic.

Bonus: the bootstrap

You don't have to do this now, but I think it's very helpful to understand sampling distributions and standard errors.

We're going to make a bootstrap estimate of the regression slope in our model:

```
boot_beta <- numeric(9999)
for (i in 1:9999) {
  boot_sample <- sample(1:nrow(study), replace = TRUE)
  x_boot <- study[boot_sample, "Athena.Raw"]
  y_boot <- study[boot_sample, "Cubiks.Raw"]
  boot_beta[i] <- cor(x_boot, y_boot) * sd(y_boot) / sd(x_boot)
}
hist(boot_beta)
```

Bonus: the bootstrap

The plot shows the bootstrapped sampling distribution of the slope.

The distribution is centred around the estimated parameter:

```
> mean(boot_beta)
[1] 1.130819
```

The sd is an approximation of the standard error of the estimate:

```
> sd(boot_beta)
[1] 0.1272132
```

