Simple linear regression

Where we're headed

- Mixing interval and categorical variables
- Modeling dependent variables as a function of multiple independent variables
- Studies with *repeated measures* (e.g., multiple samples per subject or item)

Outline for the next few weeks

- (Today) *simple linear regression*
- And in the near future:
  - more complex linear regression, and *analysis of covariance*
  - *logistic regression*
  - *ordinal regression*
  - *isotonic regression*
  - *mixed effects regression*
  - *post-hoc tests*
- And if time allows, a final lecture on data visualization in R.

Problems addressed (1/)

Recall my (re)definition of the two-sample *t*-test from last week:

- Samples are: pairs *c*, *d* where *c* is a scalar and *d* is a boolean indicating group membership
- Null hypothesis: the two samples have the same means

Or, in other words, we have an interval DV (*c*) and a binomial IV (*d*).

But what if *d* was a categorical variable of more than two levels? E.g.: "autism spectrum disorder", "specific language impairment", "typical development". One might want to formulate null hypotheses such as "all the group means are equal" ("omnibus test") or "no two pairs of group means are different" ("post-hoc test").

Problems addressed (2/)

So far we've seen:

- Categorical DV, one binomial IV: Fisher exact test
- Rank DV, one binomial IV: two-sample Wilcoxon test
- Interval DV, one binomial IV: two-sample *t*-test
- interval DV, one interval IV: Pearson *r* test
- interval/rank DV, one interval/rank IV: Spearman ρ test, Kendall τ_b test

Problems addressed (3/)

Regression and ANOVA will also allow us to do:

- Interval DV, categorical IVs: *one-way ANOVA*
- Interval DV, categorical and/or interval IVs: *linear regression*
- Binomial DV, categorical and/or interval IVs: *logistic regression*
- Rank DV, categorical and/or interval IVs: *ordinal regression*
- rank DV, rank IVs: *isotonic regression*

Simple linear regression

Simple linear regression covers designs where there is one interval DV, and one or more interval or categorical IVs.

For each (pseudo-)DV we learn its *slope* with respect to the interval DV; thus we can represent the relationship using a familiar formula:

$$Y = aX + b$$

where a is the slope and b is the y-intercept (i.e., the value of Y when $X = 0$). Thus the model is very similar to that of Pearson's r, as we'll see.

The equation for the *i*th point

$$y_i = \beta_0 + \beta_1 x_i + \varepsilon_i$$

where

- y_i is a *i*th value of the DV,
- β_0 is the *y*-intercept of the line,
- β_1 is the slope of the line,
- x_i is the *i*th value of the IV, and
- ε_i is the deviance of the *i*th observation.

Sometimes we write B = {β_0, β_1, ...}.

The equation in general

$$Y = \beta_0 + \beta_1 X + \varepsilon$$

where

- Y is an n-length vector of DV values,
- β_0 is a scalar, the y-intercept of the line,
- β_1 is a scalar, the slope of the line,
- X is an n-length vector of IV values, and
- ε is an n-length vector of deviances.

The prediction for the *i*th point

$$\hat{y}_i = \beta_0 + \beta_1 x_i$$

where

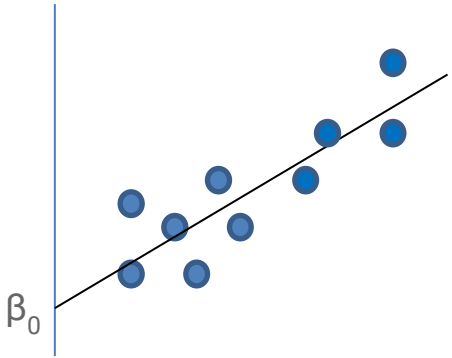
- $\hat{y}_i$ is the predicted *i*th *y*-value,
- β_0 is the *y*-intercept of the line,
- β_1 is the slope of the line, and
- x_i is the *i*th value of the IV.

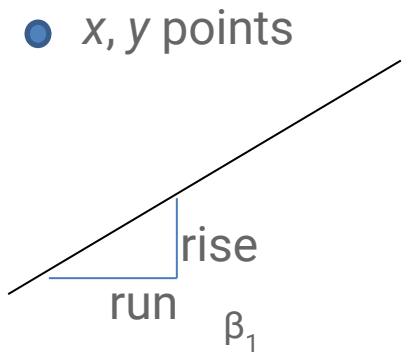
The prediction in general

$$\hat{Y} = \beta_0 + \beta_1 X$$

where

- $\hat{Y}$ is an n-length vector of predicted DV values,
- β_0 is a scalar, the y-intercept of the line,
- β_1 is a scalar, the slope of the line, and
- X is an n-length vector of IV values.


β_0


x, *y* points
rise
run
β_1

Some questions we can ask (1/)

- What is the total variance of *Y*?

This is measured by the sum of squared deviations (*SS*) of each observed y_i from the sample mean $\bar{Y}$.

$$SS_{\text{total}} = \sum_i (y_i - \bar{Y})^2.$$

Some questions we can ask (2/)

- How much variance is captured by the linear model?

This is measured by the sum of squared deviations of each prediction $\hat{y}_i$ from the sample mean $\bar{Y}$.

$$SS_{\text{model}} = \sum_i (\hat{y}_i - \bar{Y})^2.$$

Some questions we can ask (3/)

- How much variance is unmodeled (i.e., is error)?

This is measured by the sum of squared deviations of each observation y_i from the prediction $\hat{y}_i$.

$$SS_{\text{error}} = \sum_i (y_i - \hat{y}_i)^2.$$

Partitioning the sum of squared errors

In general,

$$SS_{\text{total}} = SS_{\text{model}} + SS_{\text{error}}$$

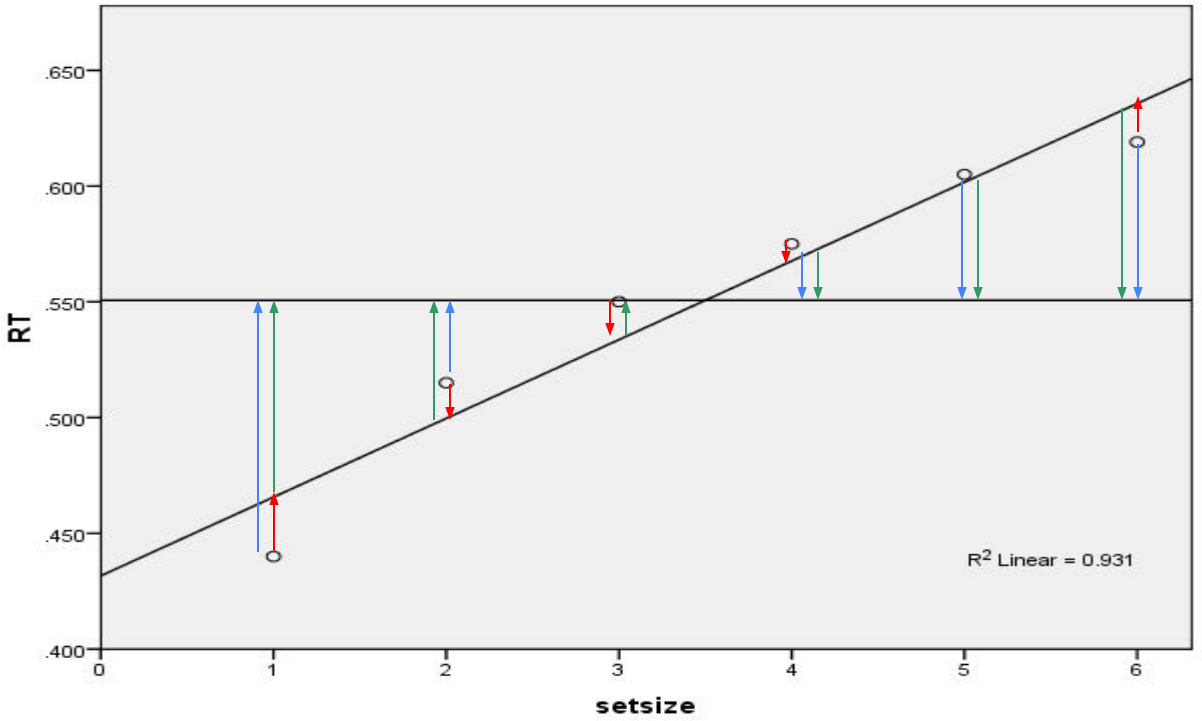
where

$$SS_{\text{total}} = \sum_i (y_i - \bar{Y})^2,$$
$$SS_{\text{model}} = \sum_i (\hat{y}_i - \bar{Y})^2, \text{ and}$$
$$SS_{\text{error}} = \sum_i (y_i - \hat{y}_i)^2.$$

$SS_{total} = \sum_i (y_i - \bar{Y})^2$

$SS_{model} = \sum_i (\hat{y}_i - \bar{Y})^2$

$SS_{error} = \sum_i (y_i - \hat{y}_i)^2$



Standard error of the estimate

Like standard deviation, this measures how much y_i differs from $\hat{y}_i$:

$$s_{Y-\hat{Y}} = \sqrt{[SS_{\text{error}} / (n - 2)]}$$

There are *n - 2* degrees of freedom here because we have to estimate two other parameters, β_0 and β_1.

Relationship to r^2

Recall that r^2 is the proportion of the total variance accounted for by the model. Therefore, it is just:

$$r^2 = SS_{model} / SS_{total}$$

Relation to *r* (1/)

If the correlation between two variables is perfect (i.e., $r = 1$ or $r = -1$), then predictability is also perfect (i.e., $\varepsilon = 0$).

In that case, if we knew the *Z*-score for some *x* we would also know the *Z*-score for *y*:

$$Z_y = +Z_x \text{ when } r = 1, \text{ and}$$
$$Z_y = -Z_x \text{ when } r = -1.$$

Relation to *r* (2/)

And even if the correlation is imperfect, our best prediction of Z_y is $r\ Z_x$:

$$\hat{Z}_y \cong r\, Z_x$$

Here, *r* is the ratios of the two variables' standard deviations; **it is the proportion of s_Y observed for each change in s_X.**

Sternberg's (1966) short-term memory scanning

Subjects are briefly shown 1-6 digits to remember (the "memory set").

After a short pause, the subject is shown a single digit (the "probe") and asked to indicate whether the digit was in the memory set or not.

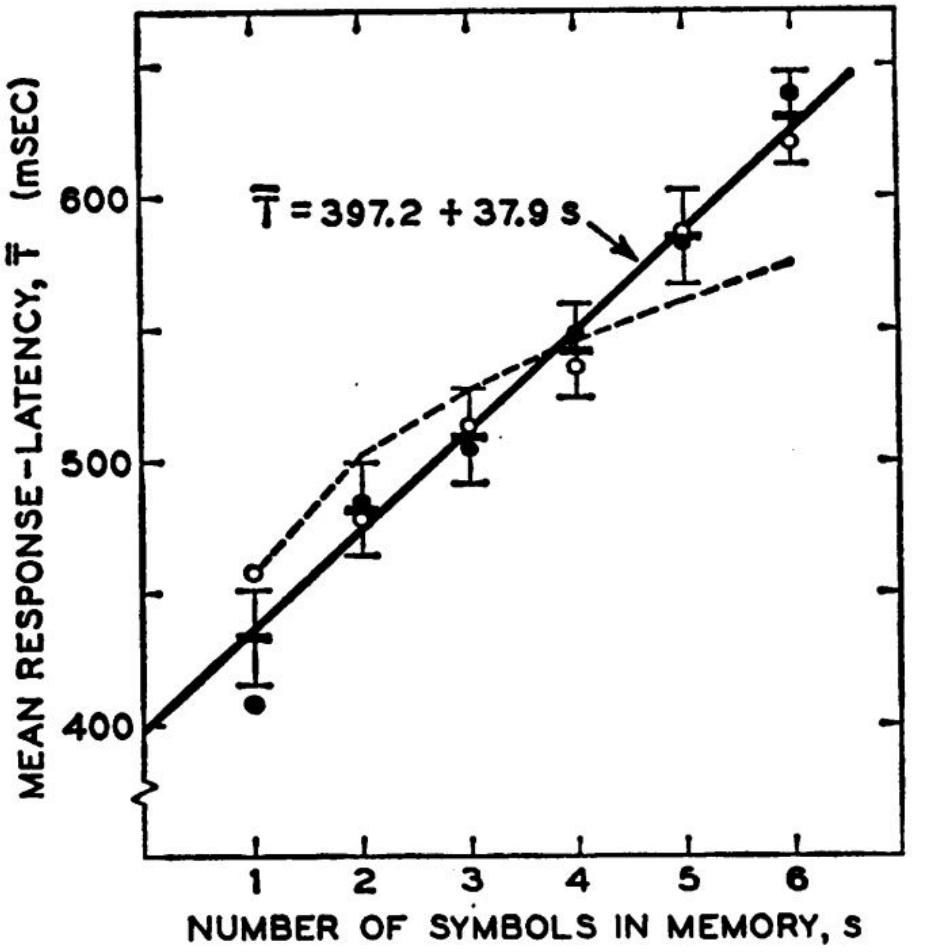
Independent variable: size of the memory set

Dependent variable: RT in seconds

Sample trial

Memory set:	3	5	2	7	9
Probe:	7				
Response:	Yes				

(Image credit: Sternberg 1966)



X	Y	$X - \bar{X}$	$Y - \bar{Y}$	$(X - \bar{X})\,(Y - \bar{Y})$
(set size)	(RT in seconds)	*X*-deviation	*Y*-deviation	Deviation prod.
1	.440	−2.5	−.110	0.2750
2	.515	−1.5	−.035	0.0525
3	.550	−0.5	.000	0.0
4	.575	0.5	.025	0.0125
5	.605	1.5	.055	0.0825
6	.615	2.5	.065	0.1625
		S_X = 1.87	**S_Y = 0.65**	**∑ = 0.585**

$\text{cov}(X, Y) = 0.585 / (6 - 1) = 0.117$

$r = \text{cov}(X, Y) / (S_X S_Y) = 0.117 / (1.87 \times 0.65) = .960$

Computing the regression model

Recall the formula

$$Y = \beta_0 + \beta_1 X + \varepsilon.$$

Then,

$$\beta_1 = r\,(s_Y / s_X)$$
$$\beta_0 = \bar{Y} - \beta_1 \bar{X}.$$

```
> x <- 1:6
> y <- c(.440, .515, .550, .575, .605, .615)
> regress <- lm(y ~ x)
> summary(regress)

Call:
lm(formula = y ~ x)

Residuals:
       1          2          3          4          5          6
-0.026429  0.015143  0.016714  0.008286  0.004857 -0.018571
...
```

```
...
Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept) 0.433000   0.018875  22.941 2.14e-05 ***
x           0.033429   0.004847   6.897  0.00232 **
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
Residual standard error: 0.02027 on 4 degrees of freedom
Multiple R-squared:  0.9224,    Adjusted R-squared:  0.903
F-statistic: 47.57 on 1 and 4 DF,  p-value: 0.002317
```

Here the row labeled `(Intercept)` gives statistics associated with β_0 and the row labeled `x` gives statistics associated with β_1.

```
> confint(regress)
                 2.5 %     97.5 %
(Intercept) 0.38059487 0.48540513
x           0.01997218 0.04688497
```

Since neither interval contains 0, we conclude that neither β_0 nor β_1 is likely to be 0 at the population level.

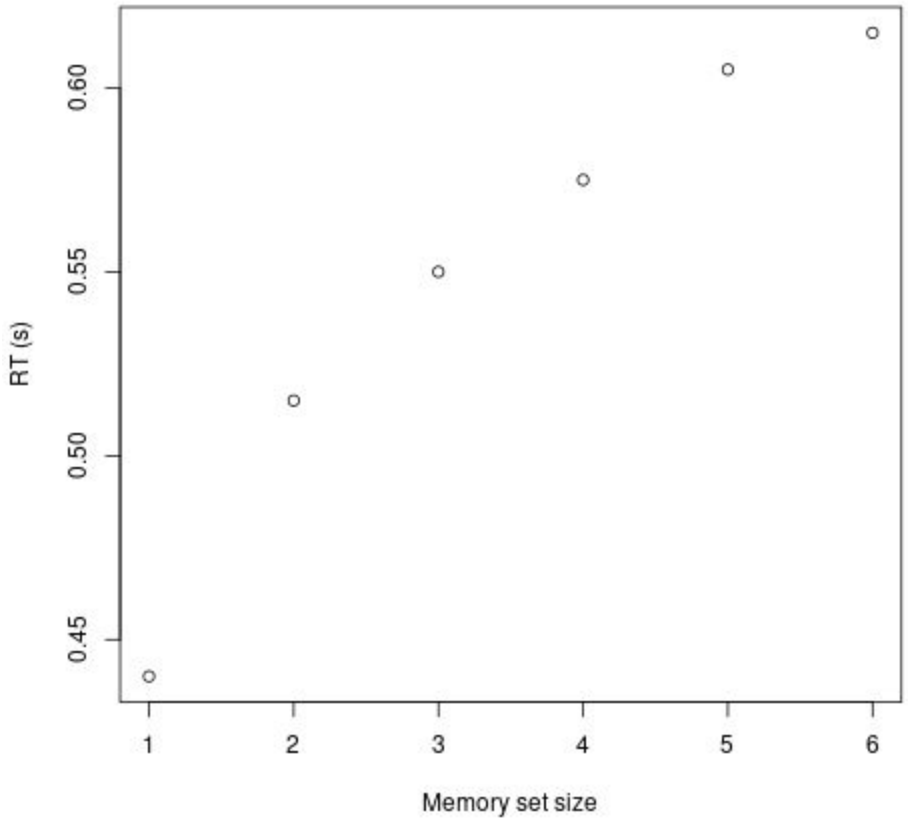
```
> fitted(regress)
        1         2         3         4         5         6
0.4664286 0.4998571 0.5332857 0.5667143 0.6001429 0.6335714
```

AKA: $\hat{Y}$.

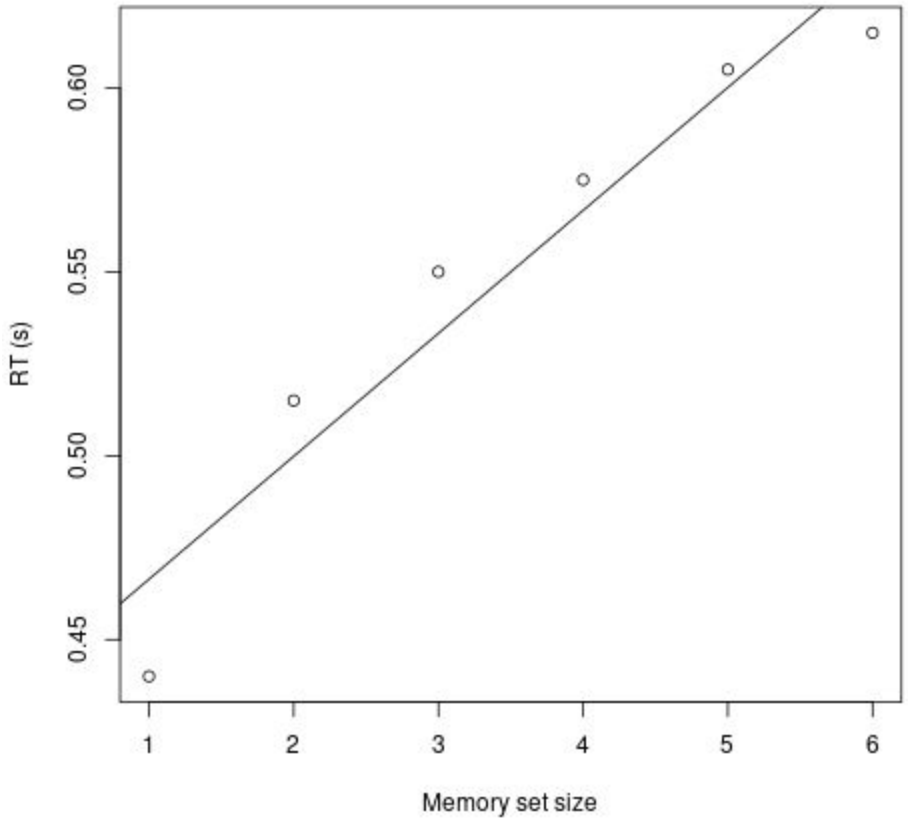
```
> residuals(regress)
           1            2            3            4
5            6
-0.026428571  0.015142857  0.016714286  0.008285714
0.004857143 -0.018571429
```

AKA: ε.

```
> plot(x, y, xlab = "Memory set size", ylab = "RT (s)")
```



```
> plot(x, y, xlab = "Memory set size", ylab = "RT (s)")
> abline(regress)
```



Using the formula to predict

- What RT (s) would we predict for set size $x = 7$?

$$\hat{y} = 0.433 + 033 \times 7 = 0.664$$

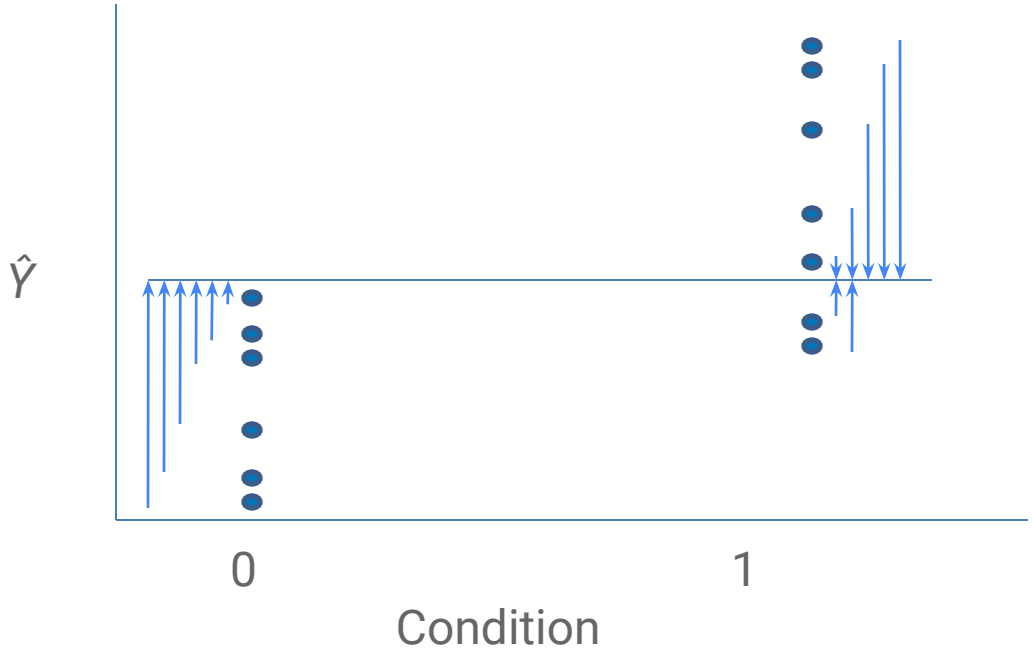
- What about for set size $x = 15$?

While the formula gives an answer, it is probably dangerous to *extrapolate* this far, since it is independently known that people generally can't remember 15-digit numbers without extensive practice.

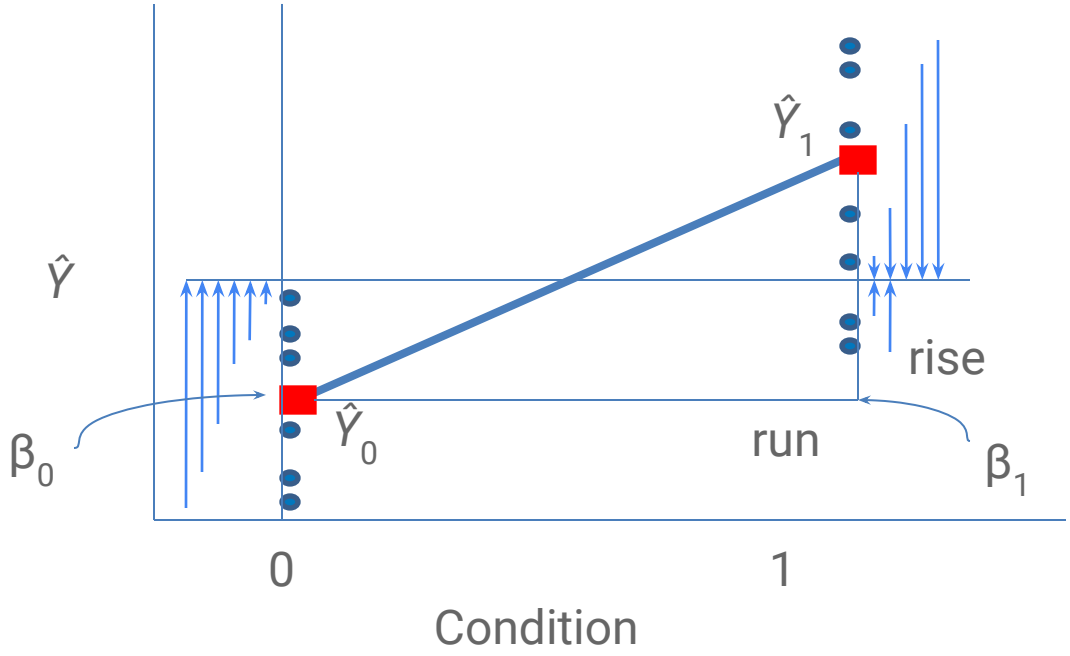
Relation to point-biserial correlation

So far we've focused on the case with a single interval IV. But just as Pearson correlation can be applied to a binomial IV (as *point-biserial correlation*), so to can linear regression.

The trick here is *dummy coding* the IV so that one level is 0 and another is 1. (This is the same trick we used last week.)



SS_{total}



SS_{total}

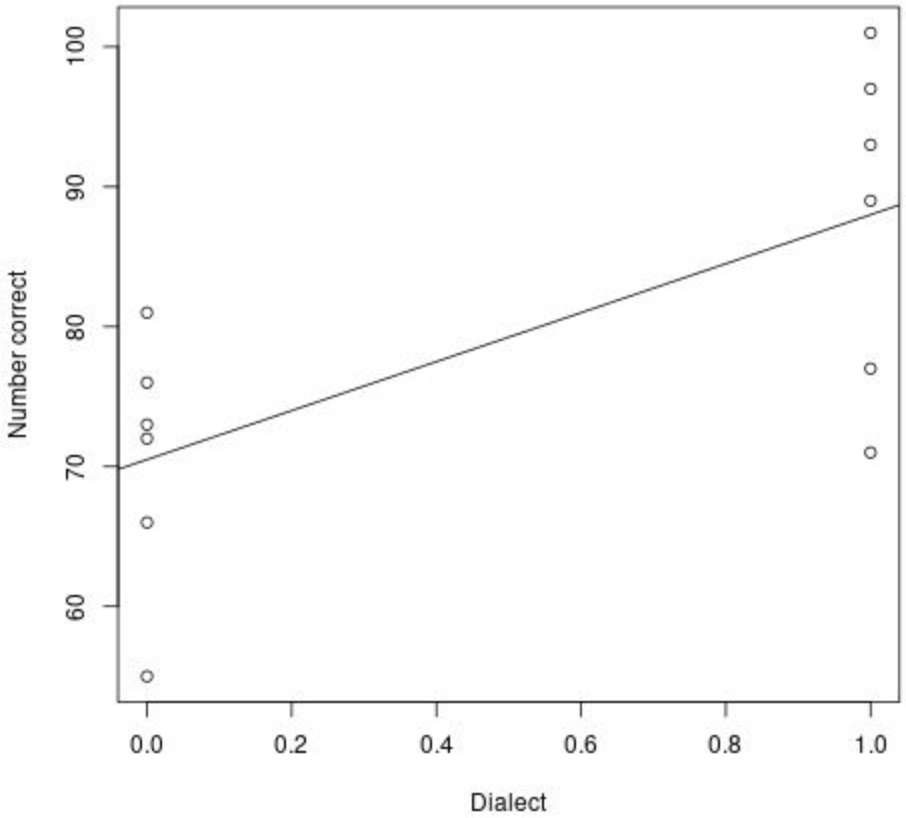
Condition mean

```
> dialect.perception <- data.frame(
+     dialect = c("A", "A", "A", "A", "A", "A",
+                 "B", "B", "B", "B", "B", "B"),
+     dialect.code = c(0, 0, 0, 0, 0, 0, 1, 1, 1, 1, 1, 1),
+     accuracy = c(55, 73, 76, 66, 72, 81, 71, 93, 77, 89,
+     101, 97))


dialect_perception <- data.frame(
     dialect = c("A", "A", "A", "A", "A", "A",
                 "B", "B", "B", "B", "B", "B"),
    dialect_code = c(0, 0, 0, 0, 0, 0, 1, 1, 1, 1, 1, 1),
    accuracy = c(55, 73, 76, 66, 72, 81, 71, 93, 77, 89,
    101, 97))
```

```
> dialect.perception
   dialect dialect.code accuracy
1        A            0       55
2        A            0       73
3        A            0       76
4        A            0       66
5        A            0       72
6        A            0       81
7        B            1       71
8        B            1       93
9        B            1       77
10       B            1       89
11       B            1      101
12       B            1       97
```

```
> plot(accuracy ~ dialect.code, data = dialect.perception,
+      xlab = "Dialect", ylab = "Number correct")
> abline(lm(accuracy ~ dialect.code,
+           data = dialect.perception))
```


Number correct
100
90
80
70
60
0.0
0.2
0.4
0.6
0.8
1.0
Dialect

As Student's *t*-test

```
> t.test(accuracy ~ dialect.code, data = dialect.perception,
+         var.equal = TRUE)

    Two Sample t-test

data:  accuracy by dialect_code
t = -2.896, df = 10, p-value = 0.01594
alternative hypothesis: true difference in means is not equal
to 0
...
```

As a linear regression

```
> regress <- lm(accuracy ~ dialect,
+               data = dialect.perception)
> summary(regress)
...
             Estimate Std. Error t value Pr(>|t|)
(Intercept)    70.500      4.273  16.499  1.4e-08 ***
dialect_code   17.500      6.043   2.896   0.0159 *
...
```

Note that R handles the dummy coding for us so we use the categorical `dialect` rather than the numerical `dialect.code`.

Sidebar

We can also use this "formula syntax" (i.e., with ~) in many R statistics functions such as `t.test`, `wilcox.test`, and so on:

```
> iris2 <- droplevels(
+     subset(iris, Species %in% c("versicolor", "virginica")))
> t.test(Sepal.Width ~ Species, data = iris2)
        Welch Two Sample t-test

data:  Sepal.Width by Species
t = -3.2058, df = 97.927, p-value = 0.001819
...
```

Using the formula to predict

- What's the mean accuracy for dialect A ($x = 0$)?

$$\hat{y} = 70.5 + 17.5 \times 0 = 70.5$$

- What the mean accuracy for dialect B ($x = 1$)?

$$\hat{y} = 70.5 + 17.5 \times 1 = 88.0$$

The two-sample *t*-test is a special case

Indeed, virtually *all* parametric statistical tests (including ANOVA) are special cases of what is called *generalized linear regression*.

Assumptions of simple linear regression

- A linear relationship between DV and IVs
- Bivariate normality
- Samples are independent and identically distributed (modulo IVs)
- *Homoscedasticity* (e.g., homogeneity of variance); check this for a binomial IV by applying the *F*-test for homoscedasticity (`var.test`).

Note for computational linguistics students

There is a relatively close relationship between linear regression models and what speech and NLP researchers call *linear models* or *maximum entropy* (or *maxent*) models, but the estimation procedures used are different (e.g., they often work online in small batches rather than all at once).

Questions? Please take them to email, or Slack.