The Two-Factor Mixed Model

Section 13.3

Nomenclature

- Two-Factor
  - Let ***F*** and ***R*** be our factors and note there are two
- Mixed Model
  - ***F*** is a fixed factor
  - ***R*** is a random factor

Thus we have a Two-Factor Mixed Model

suppose we have a fictional fat city called McAllen, Texas

- We want to test a weight loss cake

fattest city in america

wallethub.com
https://wallethub.com › edu › fattest-cities-in-america

Most Overweight and Obese Cities in the U.S. - WalletHub

Mar 13, 2023 — **Fattest Cities** in the **U.S.** ; 1, McAllen-Edinburg-Mission, TX, 85.93 ; 2, Memphis, TN-MS-AR, 84.88 ; 3, Mobile, AL, 84.52 ; 4, Knoxville, TN, 84.31 ...

beckershospitalreview.com
https://www.beckershospitalreview.com › 10-most-ove...

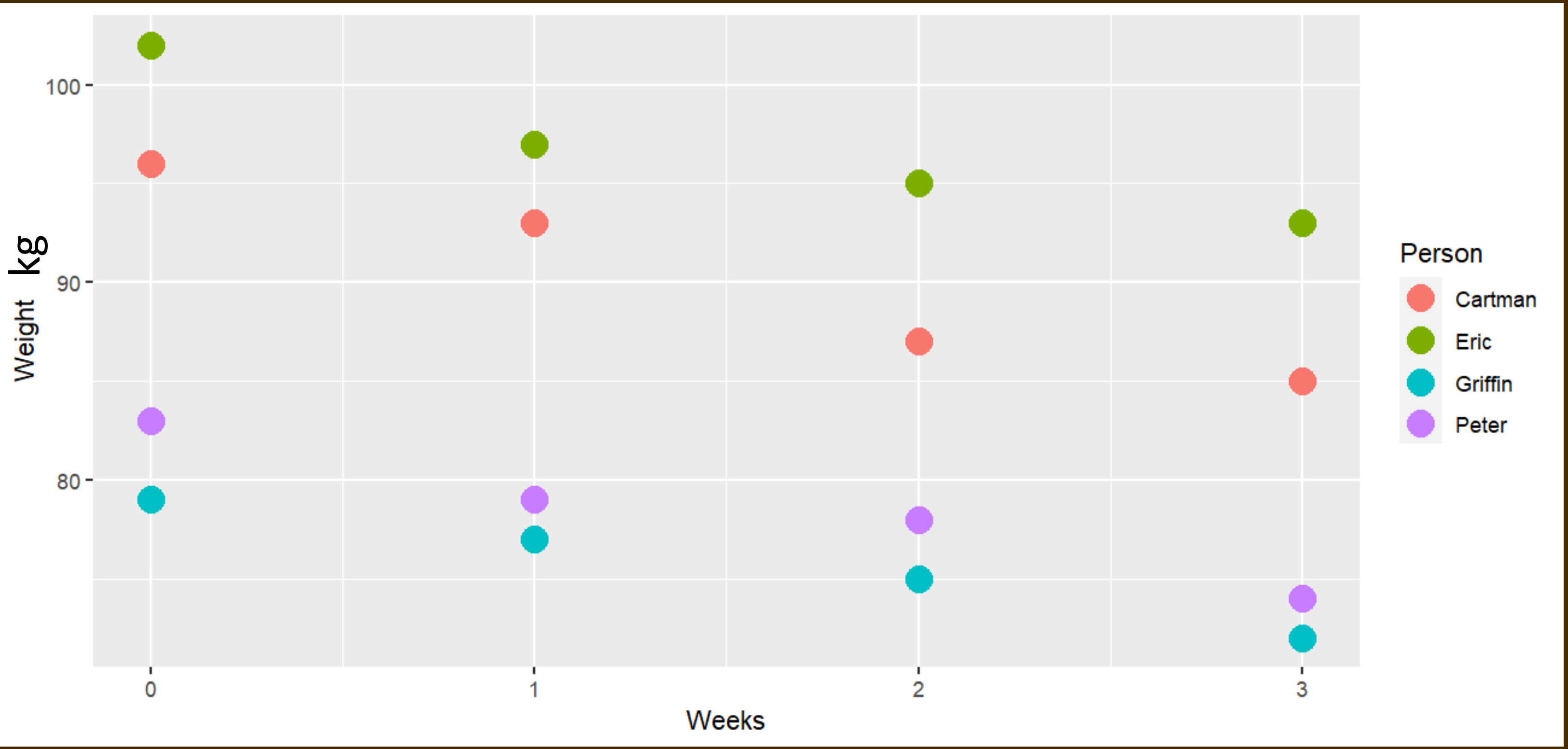
10 most overweight US cities - Becker's Hospital Review

Mar 14, 2022 — McAllen, Texas, is the most overweight and **obese city** in the **U.S.**, according to an analysis by WalletHub, a personal finance website.

Hypothesis testing

- H0: $\mu_0 = \mu_{Cake747}$
  - Cake recipe 747 diet does not cause weight loss
- Ha: $\mu_0 > \mu_{Cake747}$
  - Cake recipe 747 in diet causes significant weight loss

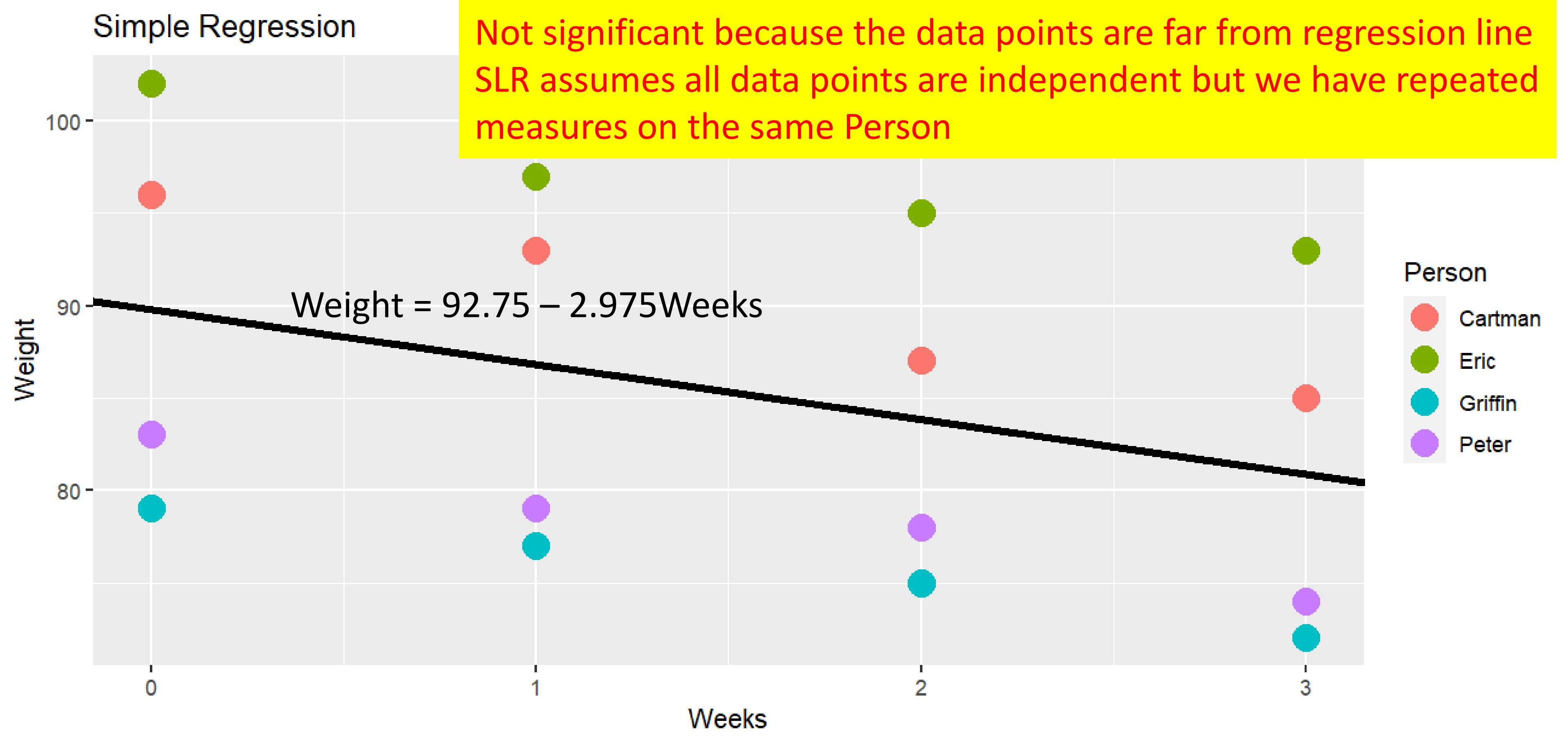
Data



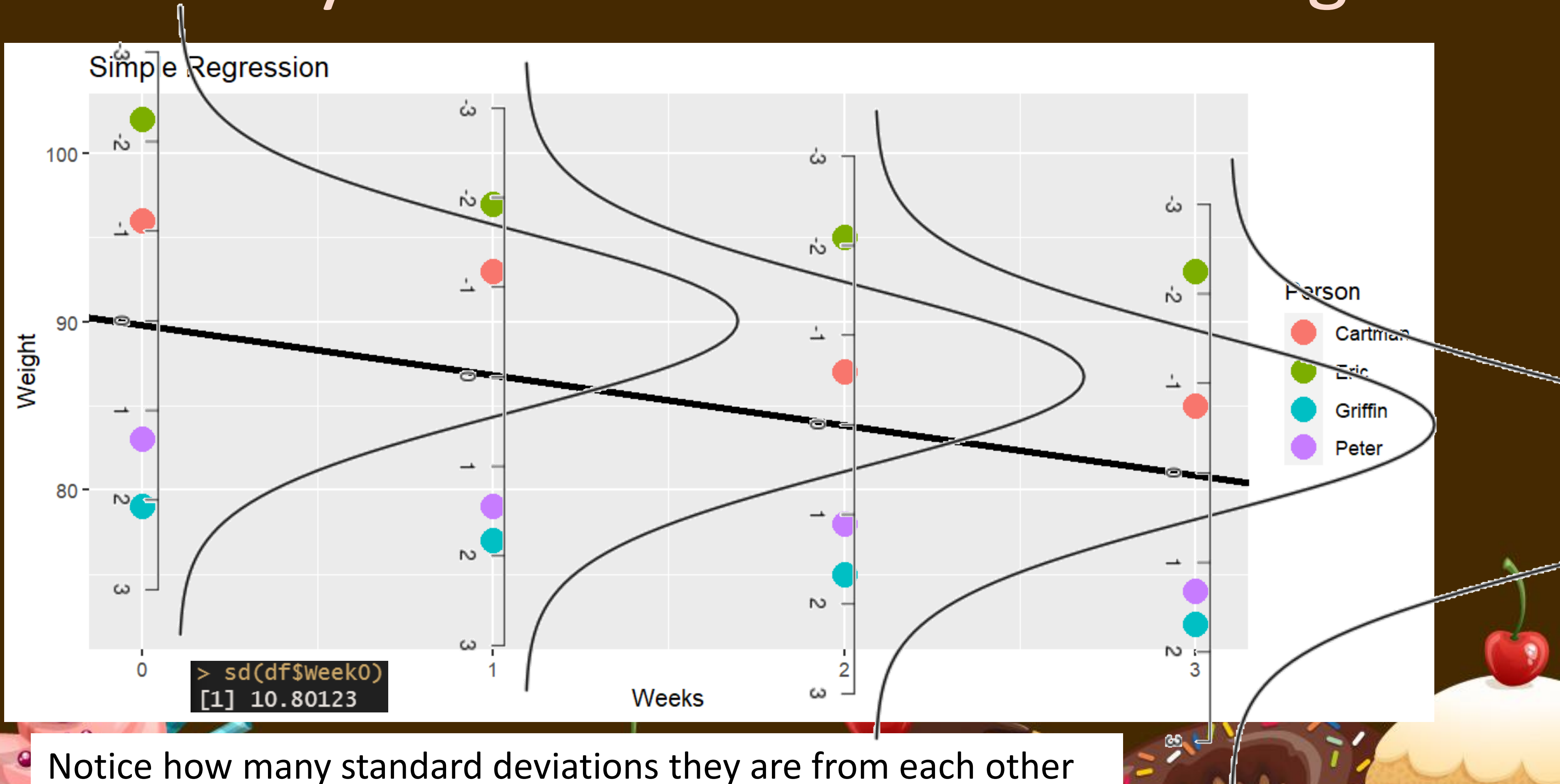

```r
df <- data.frame(
  Person = c("Eric", "Cartman", "Peter", "Griffin"),
  Diet = c(747, 747, 0, 0),
  Week0 = c(102, 96, 83, 79),
  Week1 = c(97, 93, 79, 77),
  Week2 = c(95, 87, 78, 75),
  Week3 = c(93, 85, 74, 72)
)

dfLong <- reshape(df, varying = c("Week0", "Week1", "Week2", "Week3"),
                  v.names = "Weight",
                  direction = "long",
                  timevar = "Weeks")
```

```
Call:
lm(formula = Weight ~ Weeks, data = dfSimpleRegression)

Residuals:
    Min      1Q  Median      3Q     Max 
-10.775  -8.056  -1.325   7.219  12.225 

Coefficients:
            Estimate Std. Error t value Pr(>|t|)    
(Intercept)   92.750      5.637  16.453 1.49e-10 ***
Weeks         -2.975      2.058  -1.445     0.17    
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘

Residual standard error: 9.206 on 14 degrees of freedom
Multiple R-squared:  0.1298, Adjusted R-squared:  0.06767 
F-statistic: 2.089 on 1 and 14 DF,  p-value: 0.1704
```

```r
library("ggplot2")
ggp <- ggplot(dfLong,aes(
  x=Weeks,
  y=Weight,
  group=Person))+geom_point(aes(color=Person), size=5)


dfLong.lm <- lm(Weight ~ Weeks, dfLong)
summary(dfLong.lm)
coeff<-coefficients(dfLong.lm)
intercept<-coeff[1]
slope<- coeff[2]
ggp = ggp + ggtitle("Simple Regression")
+ geom_abline(intercept = intercept, slope = slope, color="red", size=1.5)
```



Mistake is the 4 Person randomly selected had different weights



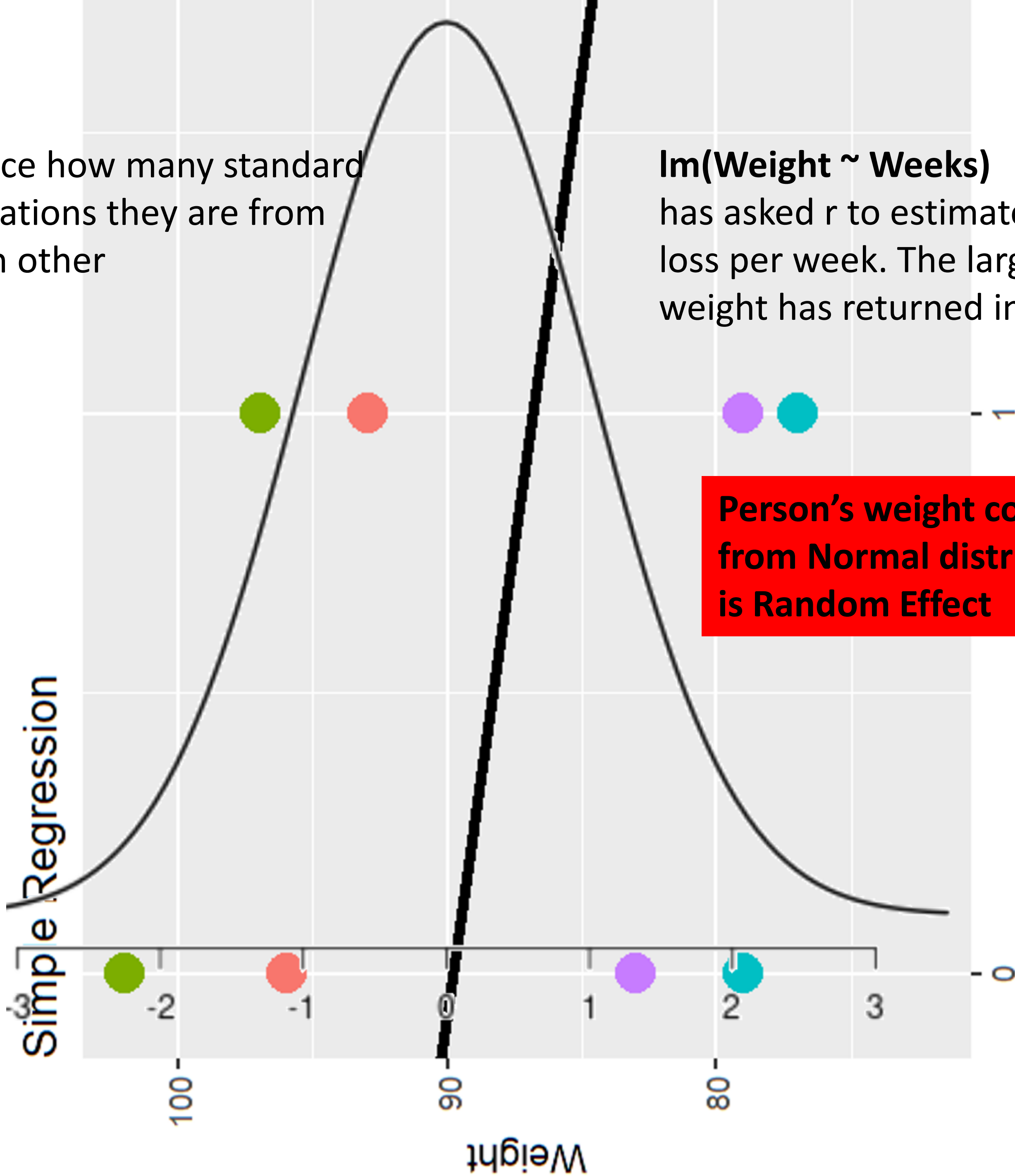
Notice how many standard deviations they are from each other

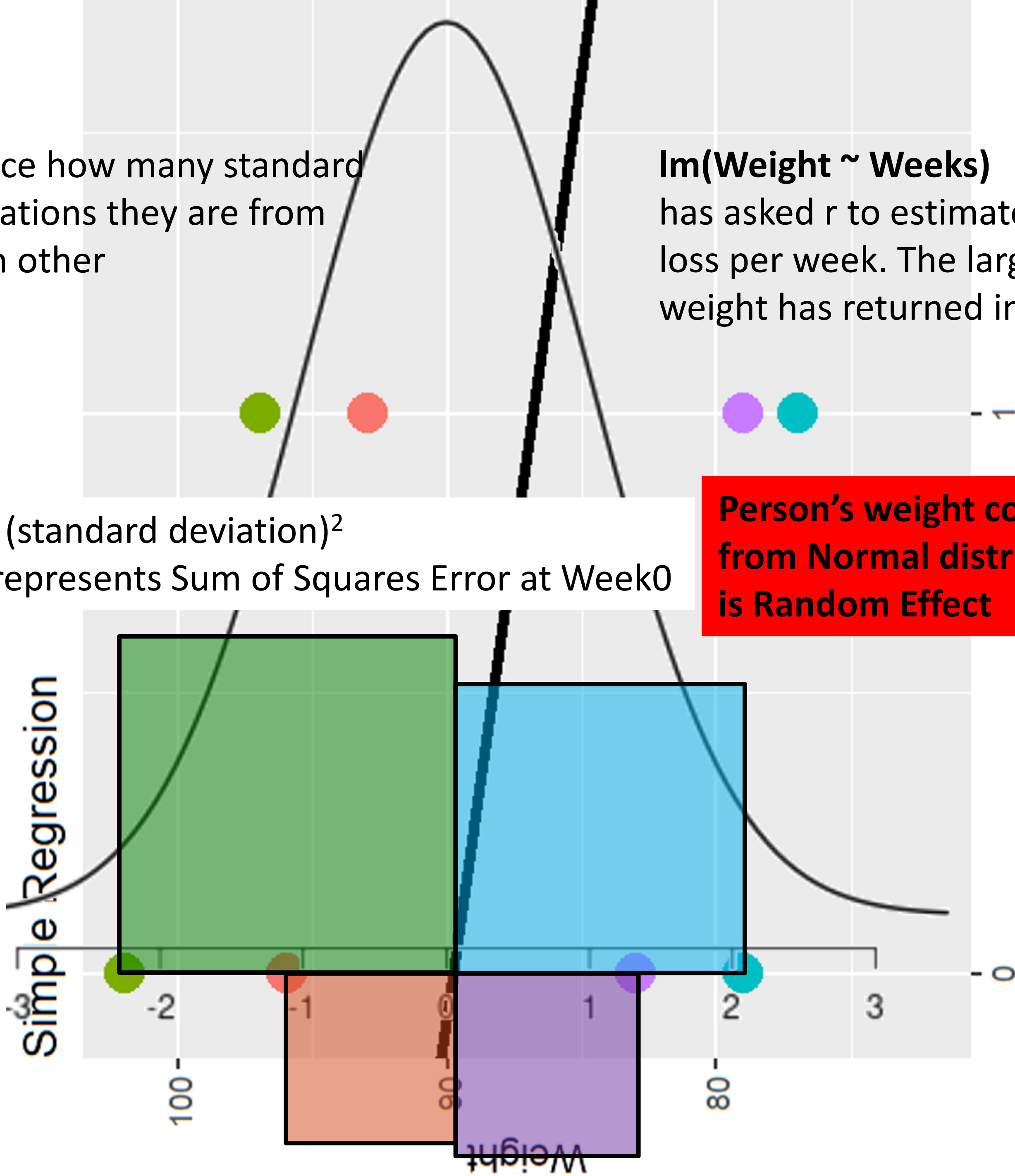
Simple linear regression is the wrong model to use. Enter **Two-Factor Mixed Model**

Notice how many standard deviations they are from each other

lm(Weight ~ Weeks)
has asked r to estimate the overall weight loss per week. The large variations in ALL weight has returned insignificant p-value.

Person’s weight comes from Normal distribution is Random Effect





Notice how many standard deviations they are from each other

lm(Weight ~ Weeks)
has asked r to estimate the overall weight loss per week. The large variations in ALL weight has returned insignificant p-value.

Variance is (standard deviation)^2
Total area represents Sum of Squares Error at Week0

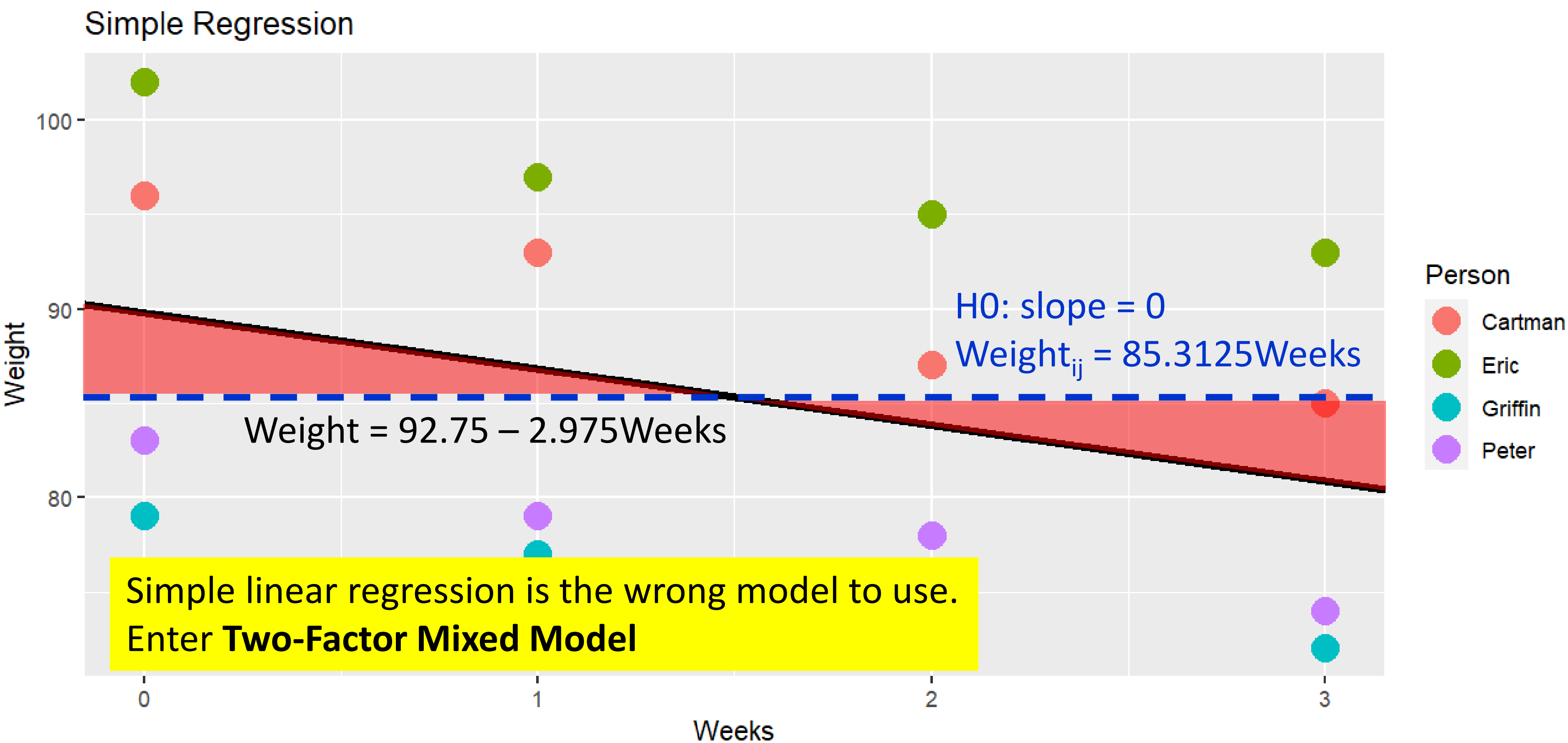
Person's weight comes from Normal distribution is Random Effect

lm(Weight ~ Weeks) has asked R to estimate the overall **weight** loss per **week**.
lm has estimated our population mean **weight** before diet to be 92.75kg
with slope negative 2.975kg loss per **week** but is insignificant relative to variation in data

```
> mean(dfLong$Weight)
[1] 85.3125
```

It has a null hypothesis that our population mean **Weight** before diet to be 85.3125kg
and the slope is zero. That is, any given **Weeks** has an estimated **Weight** of 85.3125kg for any **Person**

The large variations in ALL weight has returned insignificant p-value 0.17 > α
We conclude that nobody eating Cake747 is losing weight in McAllen, Texas



Simple linear regression is the wrong model to use.
Enter **Two-Factor Mixed Model**

Definitions

- Fixed effects represent things like the population mean that do not vary. These may represent the population parameters that we like to estimate in, for example, linear regression.
- Random effects represent parameters that can vary between groups of dependent data points. For example, if we do several measurements on the same individual, the mean of these measurements can represent one estimated parameter. Each individual may then have a unique estimate.

```
dfLong.lm <- lm(Weight ~ Weeks, dfLong)
```

```
library(lmerTest)
lmer0 <- lmer(Weight~Weeks+(1|Person), data=dfLong)
summary(lmer0)
ranef(lmer0)
```

Weeks is our fixed effect
Person is our random effect
1 specifies to use random intercept for Person
ranef() give random offset from fixed effect

```
Linear mixed model fit by REML. t-tests use Satterthwaite's meth
Formula: Weight ~ Weeks + (1 | Person)
   Data: dfLong

REML criterion at convergence: 66.2

Scaled residuals:
     Min       1Q   Median       3Q      Max
-1.53483 -0.73032 -0.01984  0.67606  1.14606

Random effects:
 Groups   Name        Variance Std.Dev.
 Person   (Intercept) 97.359   9.867
 Residual              1.294   1.138
Number of obs: 16, groups:  Person, 4

Fixed effects:
            Estimate Std. Error      df t value Pr(>|t|)
(Intercept)  92.7500     4.9825  3.1000   18.61 0.000277 ***
Weeks        -2.9750     0.2544 11.0000  -11.69 1.52e-07 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Correlation of Fixed Effects:
      (Intr)
Weeks -0.128
```

Significant

```
ac <- 4.921144
ae <- 11.399612
ag <- -9.530824
ap <- -6.789933
ggp = ggp + geom_abline(intercept = intercept+ac, slope = slope, color="red", size=.5)
ggp = ggp + geom_abline(intercept = intercept+ae, slope = slope, color="green", size=.5)
ggp = ggp + geom_abline(intercept = intercept+ag, slope = slope, color="blue", size=.5)
ggp = ggp + geom_abline(intercept = intercept+ap, slope = slope, color="purple", size=.5)
```

```
> ranef(lmer0)
$Person
        (Intercept)
Cartman    4.921144
Eric      11.399612
Griffin   -9.530824
Peter     -6.789933

with conditional variances for "Person"
```

N / lmerTest / lmerTest-package: lmerTest: Tests in Linear Mixed Effects Models

lmerTest-package: lmerTest: Tests in Linear Mixed Effects Models

In lmerTest: Tests in Linear Mixed Effects Models

Description | Key Functions and Methods | Details | Author(s) | References | Examples

Description

The lmerTest package provides p-values in type I, II or III `anova` and `summary` tables for linear mixed models (`lmer` model fits cf. lme4) via Satterthwaite's degrees of freedom method; a Kenward-Roger method is also available via the pbkrtest package. Model selection and assessment methods include `step`, `drop1`, anova-like tables for random effects (`ranova`), least-square means (LS-means; `ls_means`) and tests of linear contrasts of fixed effects (`contest`).

$Weight_i = a_i + b*Weeks$

$a_{Eric} = 92.75 + 11.40 = 104.1496$

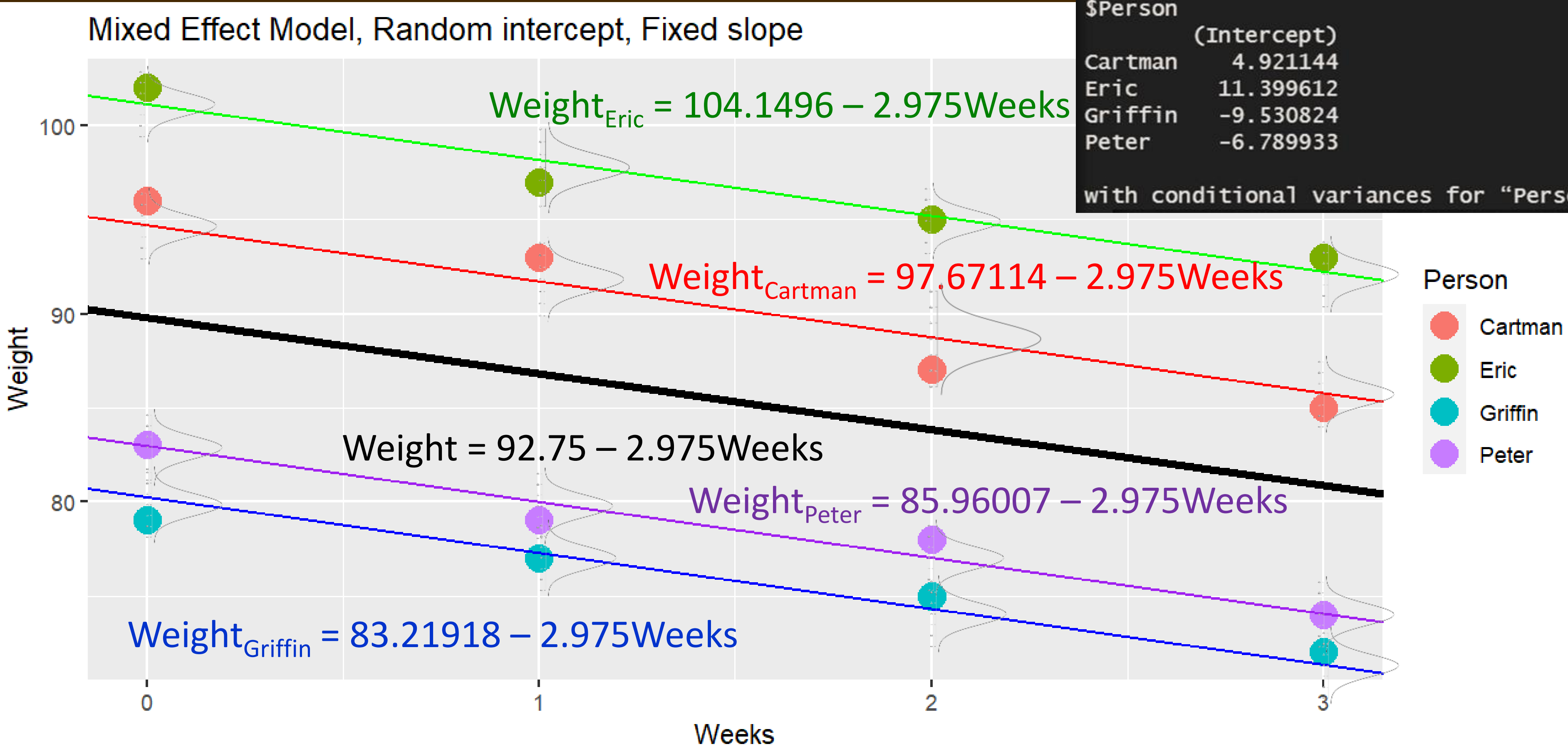
$a_{Cartman} = 92.75 + 4.9211 = 97.67114$

$a_{Peter} = 92.75 – 6.80 = 85.96007$

$a_{Griffin} = 92.75 – 9.53 = 83.21918$

Notice how small the variances are from their own lines
The advantage is that the data points are now much closer to the lines, which will result in a much smaller standard error and therefore a smaller p-value.

```
> ranef(lmer0)
$Person
        (Intercept)
Cartman    4.921144
Eric      11.399612
Griffin   -9.530824
Peter     -6.789933

with conditional variances for “Person
```



Notice how small the variances are from their own lines
Hence the drop in SE and **huge drop in SSR**

SLR

```
Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)   89.775      3.851  23.312 1.33e-12 ***
Weeks         -2.975      2.058  -1.445     0.17
```

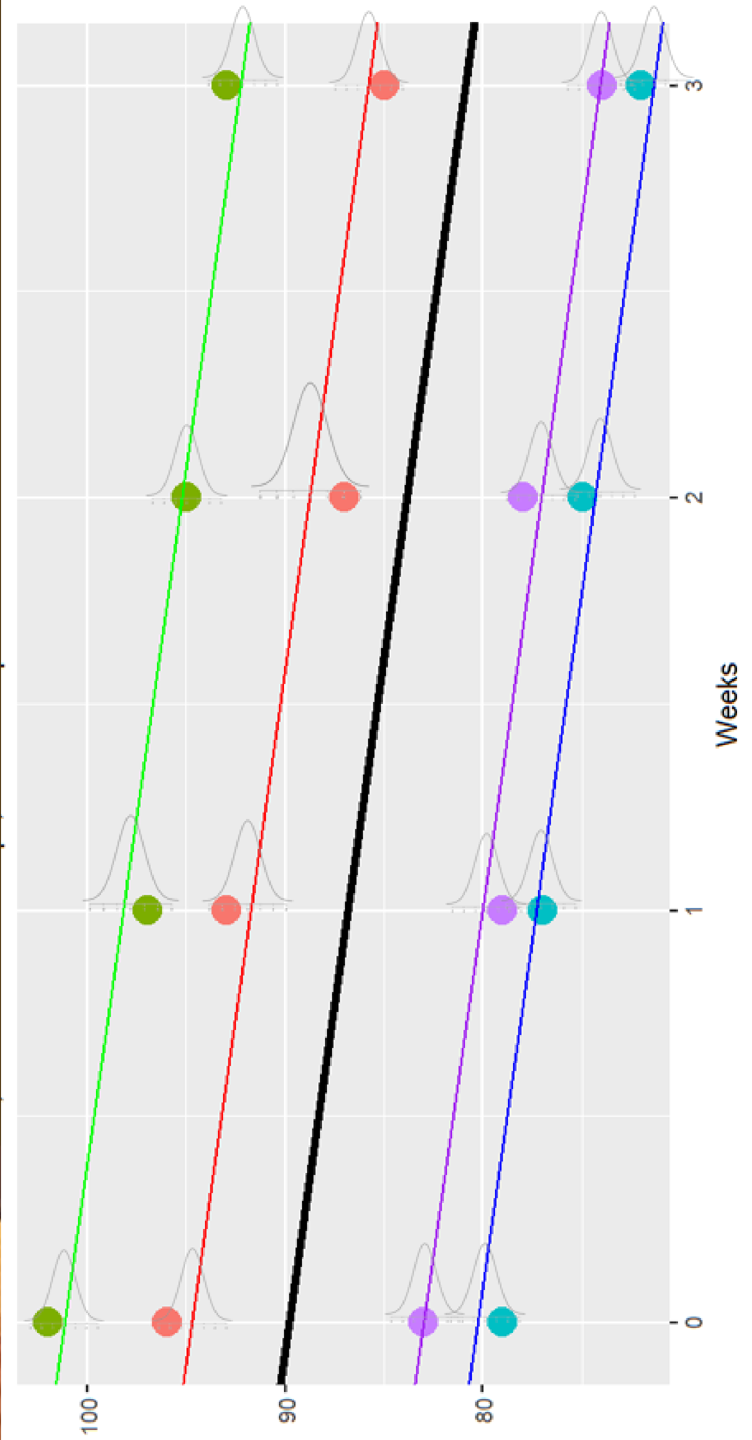
```
> summary(aov(Weight ~ Weeks, df0))
            Df Sum Sq Mean Sq F value Pr(>F)
Weeks        1    177  177.01   2.089   0.17
Residuals   14   1186   84.74
```

2FMM

```
Fixed effects:
            Estimate Std. Error      df t value Pr(>|t|)
(Intercept)  89.7750     4.9564  3.0359   18.11 0.000342 ***
Weeks        -2.9750     0.2544 11.0000  -11.69 1.52e-07 ***
```

```
> sum(residuals(lmer0)^2)
[1] 14.25036
```

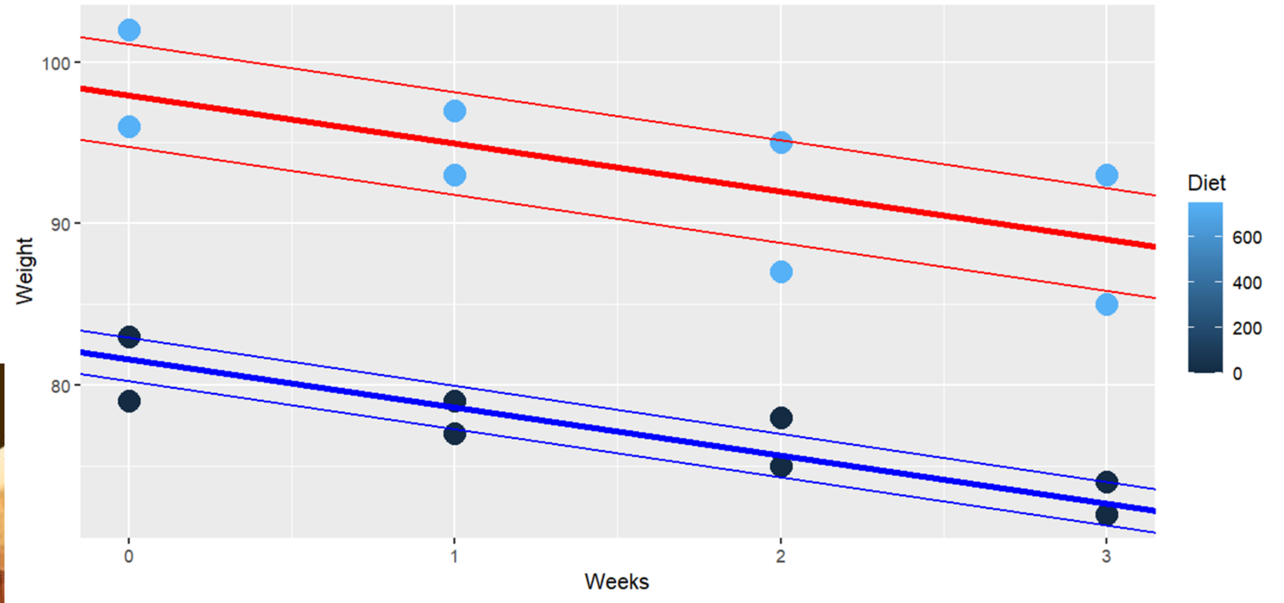
Don't know how to call SSR so manually calculated it



Hypothesis testing

- H0: $\mu_{Cake0} = \mu_{Cake747}$
  - Cake recipe 747 in diet does not cause weight loss compared to cake recipe 0
- Ha: $\mu_{Cake0} > \mu_{Cake747}$
  - Cake recipe 747 in diet causes significant weight loss compared to cake recipe 0

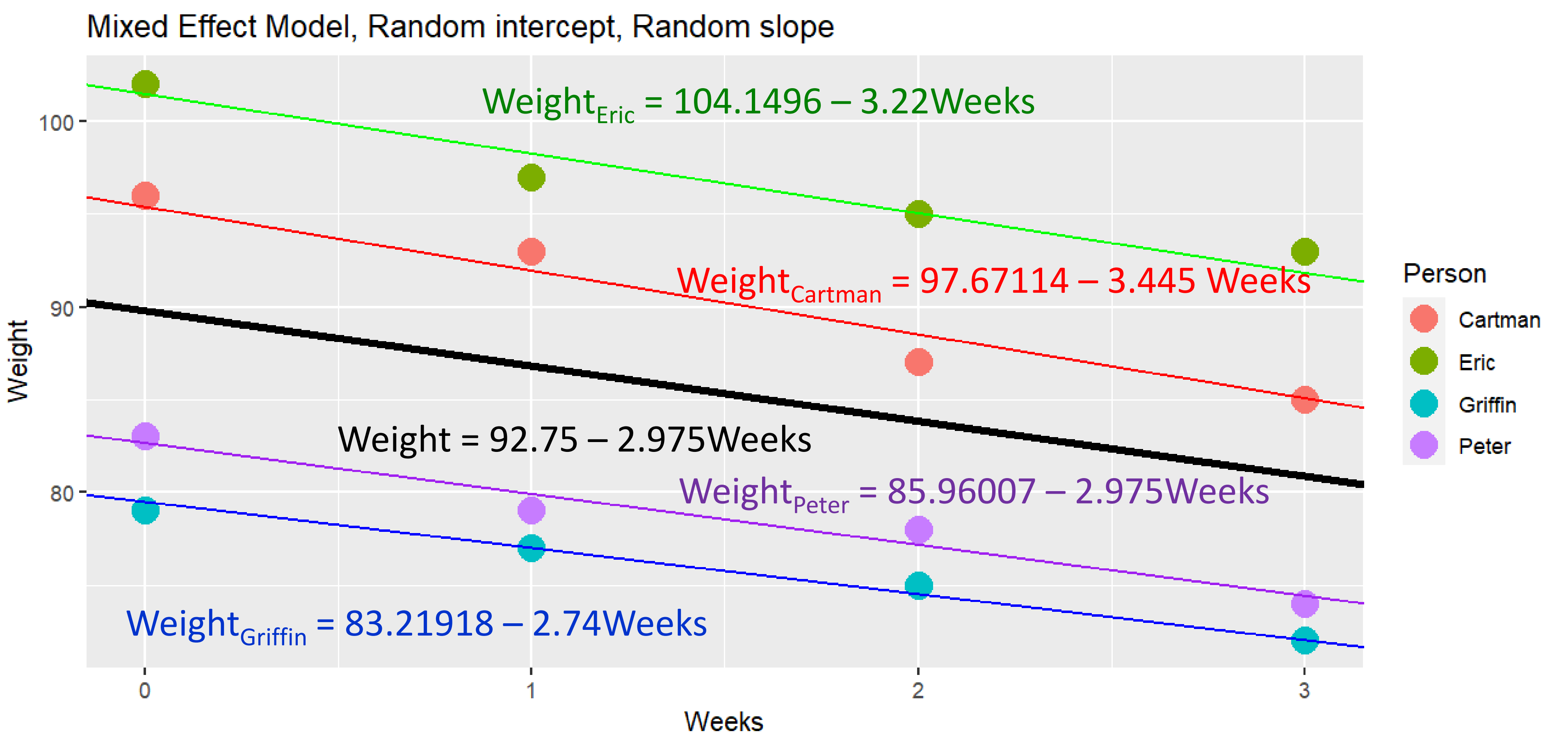
There was a different flatter slope all this time to force a higher **lm()** pvalue and be reused later in other tests.
Can you already guess the answer? Shout it out!



Random intercept & Random Slope

```r
# random intercept, fixed slope
lmer0 <- lmer(Weight~Weeks+(1|Person), data=dfLong)
# random intercept, random slope slope
lmer1 <- lmer(Weight~Weeks+(1+Weeks|Person), data=dfLong)
ranef(lmer1)
```

```
$Person
        (Intercept)      Weeks
Cartman    5.653001 -0.4702121
Eric      11.763864 -0.2462449
Griffin  -10.278128  0.4862867
Peter     -7.138737  0.2301703

with conditional variances for "Person"
```



“Weeks” P-Values so far

- Simple Linear Regression
  - ✓ 0.17
- Fixed Effect Weeks, Random Effect Person
  - Random Intercept, fixed Slope
    - ✓ 1.52e-07
  - Random Intercept, Random Slope
    - ✓ 0.003013
- Multiple Linear Regression
  - Should not be used since it violate the assumption of independence.
  - the observations are not independent because we have four data points from the same person.

When should I use Mixed Model vs another Model?

lmer will run will NA values

```
dfMissing <- dfLong
dfMissing[9,4] = NA
lmer0 <- lmer(Weight~Weeks+(1|Person), data=dfMissing)
summary(lmer0)
```

```
Linear mixed model fit by REML. t-tests use Satterthwaite's met
Formula: Weight ~ Weeks + (1 | Person)
   Data: dfMissing

REML criterion at convergence: 63.7

Scaled residuals:
     Min       1Q   Median       3Q      Max
-1.47020 -0.75226  0.02032  0.61565  1.10688

Random effects:
 Groups   Name        Variance Std.Dev.
 Person   (Intercept) 97.932   9.896
 Residual              1.415   1.189
Number of obs: 15, groups:  Person, 4

Fixed effects:
            Estimate Std. Error      df t value Pr(>|t|)
(Intercept)  92.7500     5.0013  3.1052   18.55 0.000277 ***
Weeks        -2.9675     0.2682  9.9989  -11.06 6.25e-07 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Correlation of Fixed Effects:
      (Intr)
Weeks -0.132
```

	Person	Diet	Weeks	Weight	id
1.1	Eric	747	1	102	1
2.1	Cartman	747	1	96	2
3.1	Peter	0	1	83	3
4.1	Griffin	0	1	79	4
1.2	Eric	747	2	97	1
2.2	Cartman	747	2	93	2
3.2	Peter	0	2	79	3
4.2	Griffin	0	2	77	4
1.3	Eric	747	3	NA	1
2.3	Cartman	747	3	87	2
3.3	Peter	0	3	78	3
4.3	Griffin	0	3	75	4
1.4	Eric	747	4	93	1
2.4	Cartman	747	4	85	2
3.4	Peter	0	4	74	3
4.4	Griffin	0	4	72	4

Repeated Measures ANOVA would have dropped Eric thus reducing statistical power

When should I use Mixed Model vs another Model?

lmer will run binary outcomes

- that is, dependent variable (Weight) was instead a binary count such as Heads in a coin flipping experiment
- ANOVA assumes dependent variable is continuous.

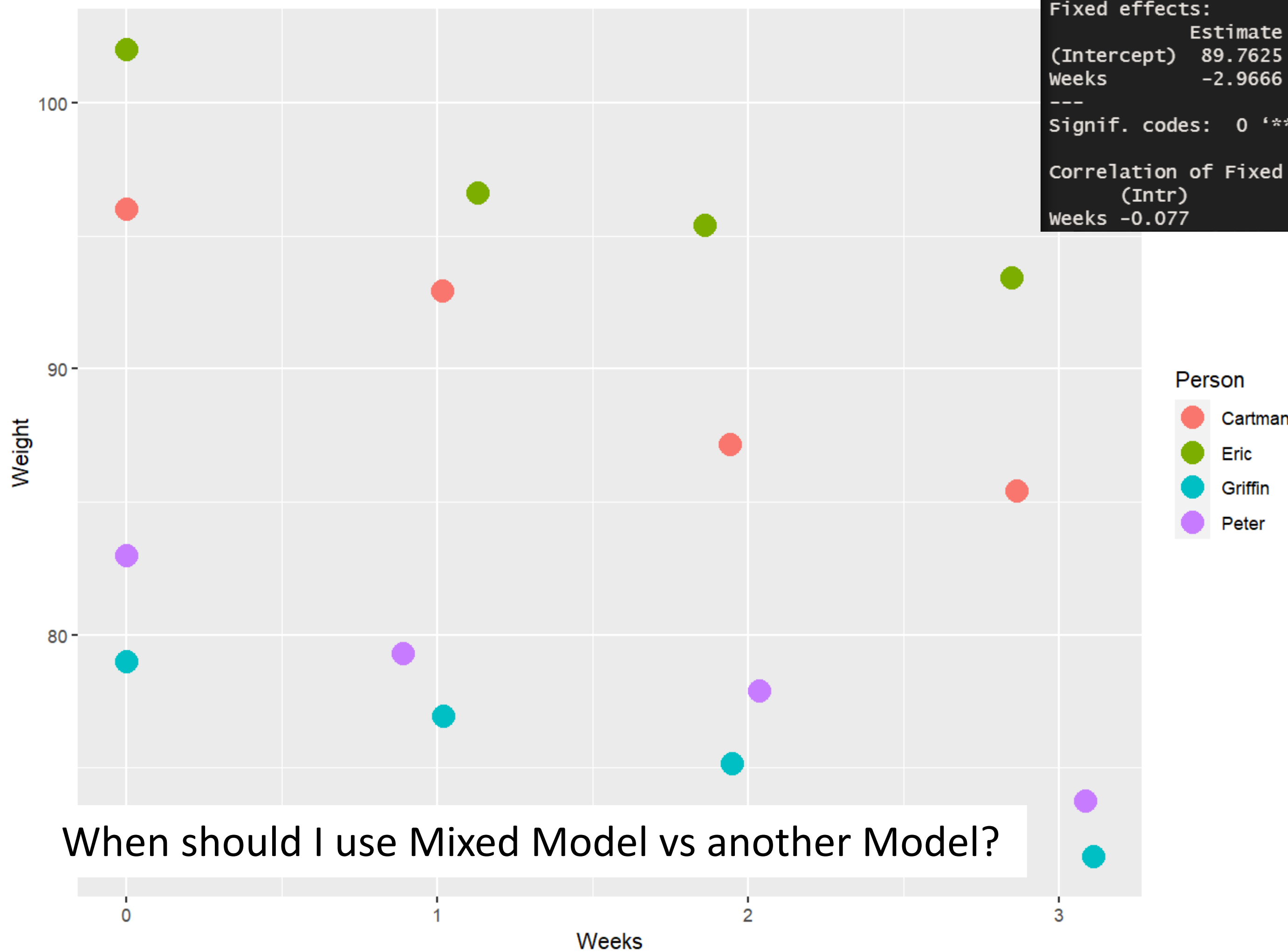
Repeated Measures ANOVA requires measures to be categorical

- That is, Weight measurements need to be on Week0, Week1, …, Week$_n$
- lmer will run using data where Weight is measured at any Week$_{\mathbb{R}}$ such as Week0, Week1.618, Week2.718, Week3.14
  - More flexibility in taking measurements
  - Simple Linear Regression will also run any real x-axis

When should I use Mixed Model vs another Model?

lmer can use any real x-axis

```
Linear mixed model fit by REML. t-tests use Satterthwaite's meth
Formula: Weight ~ Weeks + (1 | Person)
   Data: dfAnyReal

REML criterion at convergence: 66.2

Scaled residuals:
     Min       1Q   Median       3Q      Max
-1.53850 -0.73703 -0.02012  0.67110  1.15688

Random effects:
 Groups   Name        Variance Std.Dev.
 Person   (Intercept) 97.364   9.867
 Residual              1.294   1.138
Number of obs: 16, groups:  Person, 4

Fixed effects:
            Estimate Std. Error      df t value Pr(>|t|)
(Intercept)  89.7625     4.9566  3.0359   18.11 0.000342 ***
Weeks        -2.9666     0.2573 11.0000  -11.53 1.75e-07 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Correlation of Fixed Effects:
      (Intr)
Weeks -0.077
```



Weeks	Person	Diet	Weight
0.0000000	Eric	747	102.00000
0.0000000	Cartman	747	96.00000
0.0000000	Peter	0	83.00000
0.0000000	Griffin	0	79.00000
0.8910867	Peter	0	79.32402
1.0180665	Cartman	747	92.94625
1.0217975	Griffin	0	76.93515
1.1306092	Eric	747	96.61144
1.8629605	Eric	747	95.40769
1.9415453	Cartman	747	87.17390
1.9482054	Griffin	0	75.15409
2.0364102	Peter	0	77.89168
2.8480295	Eric	747	93.45211
2.8639079	Cartman	747	85.40487
3.0841013	Peter	0	73.74980
3.1122404	Griffin	0	71.66608

When should I use Mixed Model vs another Model?

Simple Linear Regression will also run any real x-axis

```
> summary(lm(Weight ~ Weeks, dfAnyReal))

Call:
lm(formula = Weight ~ Weeks, data = dfAnyReal)

Residuals:
    Min      1Q  Median      3Q     Max 
-10.577  -7.991  -1.323   7.389  12.423 

Coefficients:
            Estimate Std. Error t value Pr(>|t|)    
(Intercept)   89.577      3.253  27.536   <2e-16 ***
Weeks         -2.876      1.660  -1.733   0.0949 .  
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 8.819 on 26 degrees of freedom
Multiple R-squared:  0.1036,	Adjusted R-squared:  0.06908 
F-statistic: 3.003 on 1 and 26 DF,  p-value: 0.09493

> summary(aov(Weight ~ Weeks, dfAnyReal))
            Df Sum Sq Mean Sq F value Pr(>F)  
Weeks        1  233.6  233.62   3.003 0.0949 .
Residuals   26 2022.3   77.78                 
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
```

If we only have two Weight measurements per Person

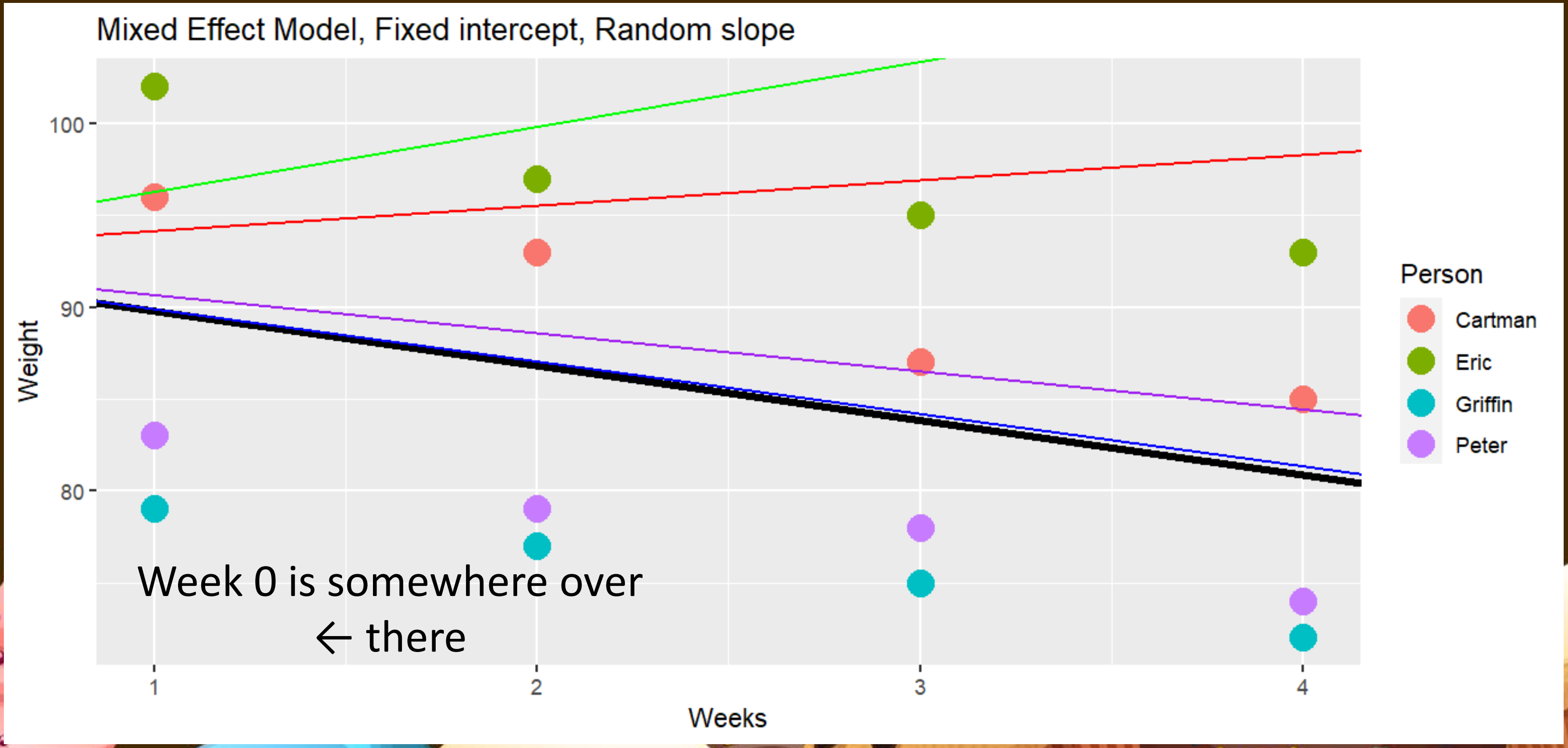
- the linear mixed-effects model with random intercepts will result in the exact same p-value as the repeated measure ANOVA and the paired t-test but only if the repeated measurements have a positive correlation

Random Slope only

Reshape started Weeks at 1

```
# random intercept, fixed slope
lmer0 <- lmer(Weight~Weeks+(1|Person), data=dfLong)
# random intercept, random slope slope
lmer1 <- lmer(Weight~Weeks+(1+Weeks|Person), data=dfLong)
# fixed intercept, random slope
lmer2 <- lmer(Weight~Weeks+(0+Weeks|Person), data=dfLong)
```

```
dfLong <- reshape(df, varying = c("Week0", "Week1", "Week2", "Week3"),
                  v.names = "Weight",
                  direction = "long",
                  timevar = "Weeks")
# fixes reshape starting at 1
library(dplyr)
df0 <- dfLong %>% mutate(across(Weeks, ~  . - 1))
```

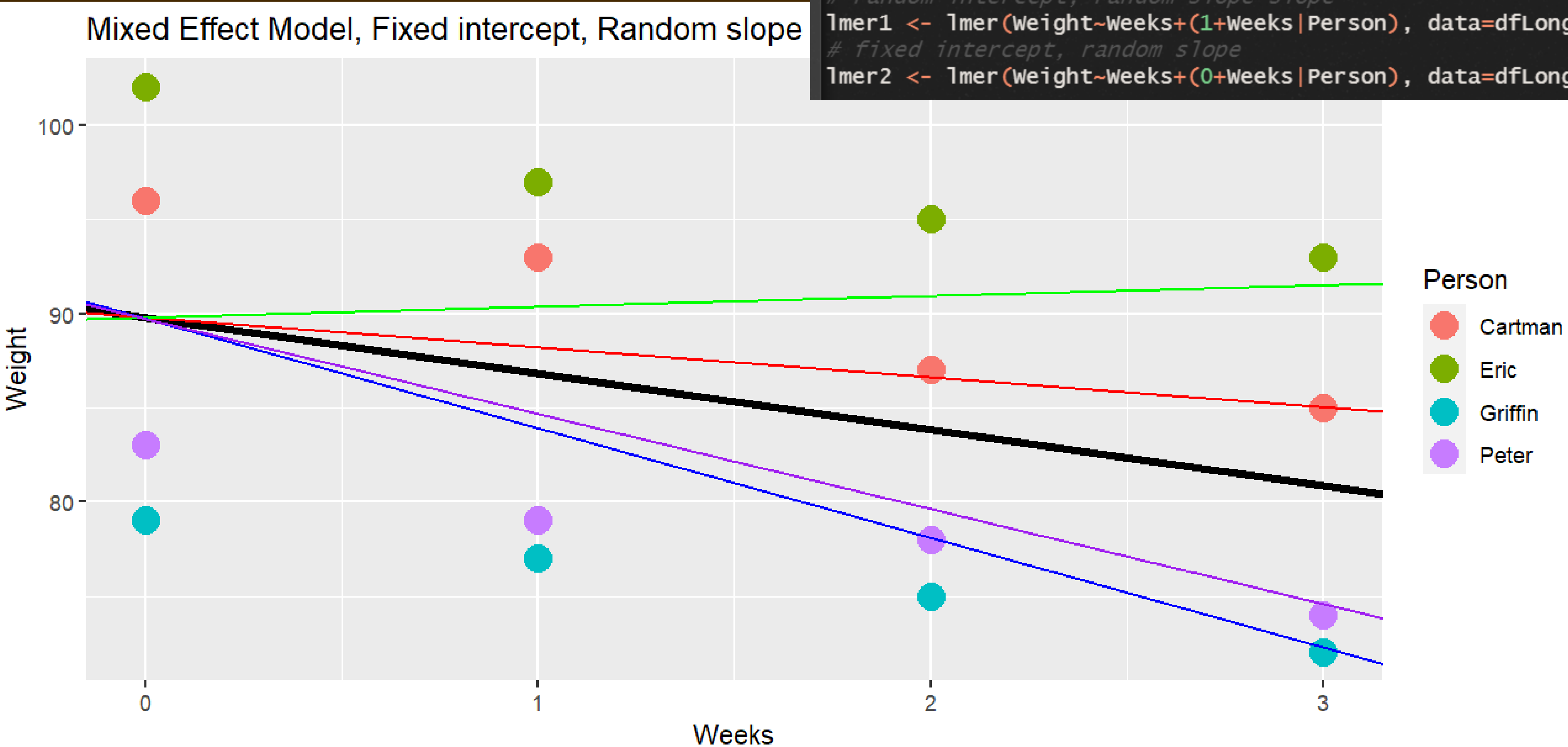


Back to how to use lmer

Random Slope, Fixed intercept

- This looks wrong
- Transform our data to use Fixed Intercept, Random Slope

```r
# random intercept, fixed slope
lmer0 <- lmer(Weight~Weeks+(1|Person), data=dfLong)
# random intercept, random slope slope
lmer1 <- lmer(Weight~Weeks+(1+Weeks|Person), data=dfLong)
# fixed intercept, random slope
lmer2 <- lmer(Weight~Weeks+(0+Weeks|Person), data=dfLong)
```

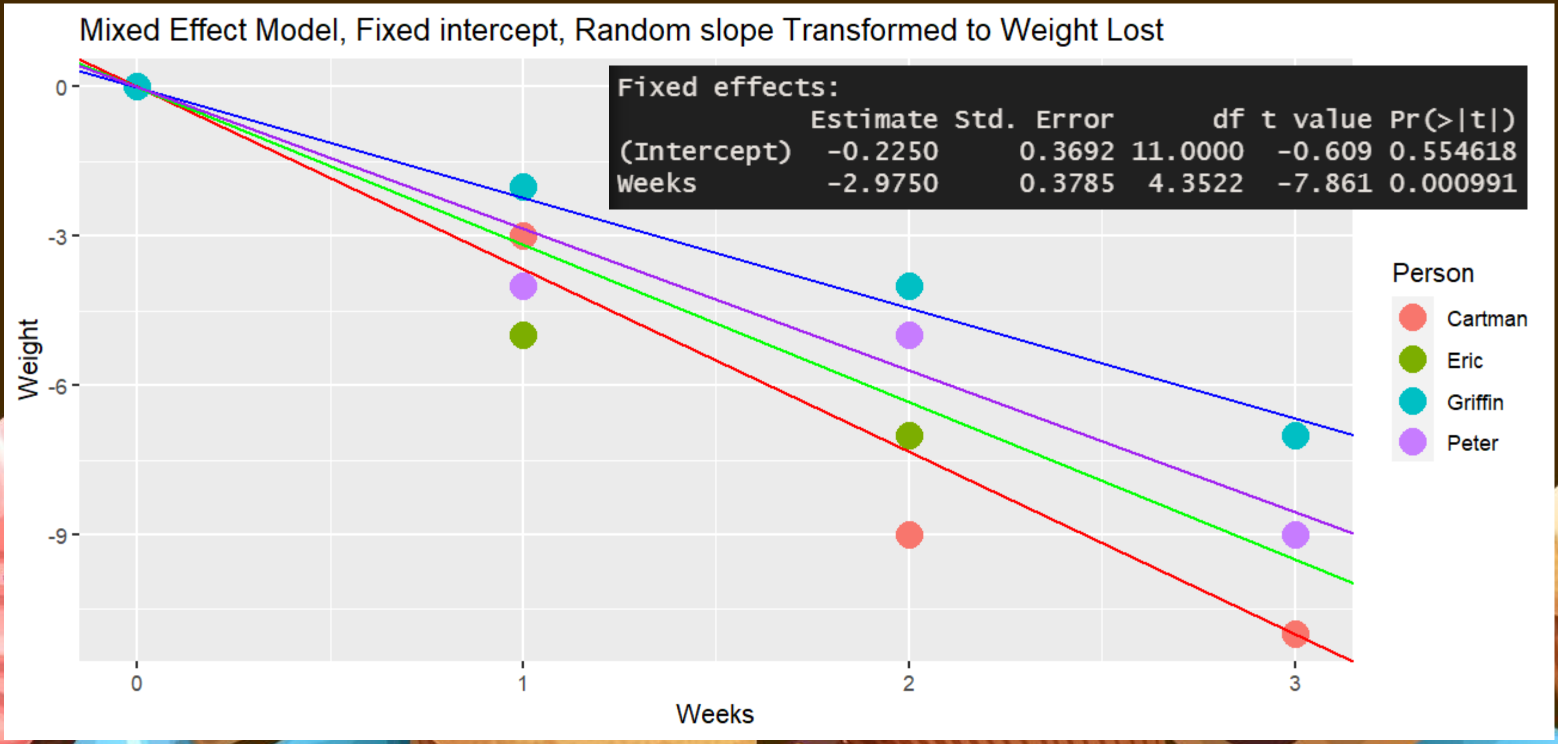


Random Slope, Fixed intercept

```
weightLoss <- df
for(r in 1:4){
  for(c in 6:3) {
    weightLoss[r, c] = weightLoss[r, c] - weightLoss[r,3]
  }
}
```

Get difference from right to left is most efficient

	Person	Diet	Week0	Week1	Week2	Week3
1	Eric	747	0	-5	-7	-9
2	Cartman	747	0	-3	-9	-11
3	Peter	0	0	-4	-5	-9
4	Griffin	0	0	-2	-4	-7



Random Slope Fixed intercept

Running SLR on Weight Loss
would have worked from the start

```
}
```

```
> weightLoss.lm <- lm(Weight ~ Weeks, weightLossLong0)
> summary(weightLoss.lm)

Call:
lm(formula = Weight ~ Weeks, data = weightLossLong0)

Residuals:
    Min      1Q  Median      3Q     Max 
-2.8250 -0.8063  0.2125  0.4625  2.1750 

Coefficients:
            Estimate Std. Error t value Pr(>|t|)    
(Intercept)  -0.2250     0.5961  -0.377    0.711    
Weeks        -2.9750     0.3186  -9.337 2.17e-07 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 1.425 on 14 degrees of freedom
Multiple R-squared:  0.8616,	Adjusted R-squared:  0.8518 
F-statistic: 87.18 on 1 and 14 DF,  p-value: 2.169e-07
```

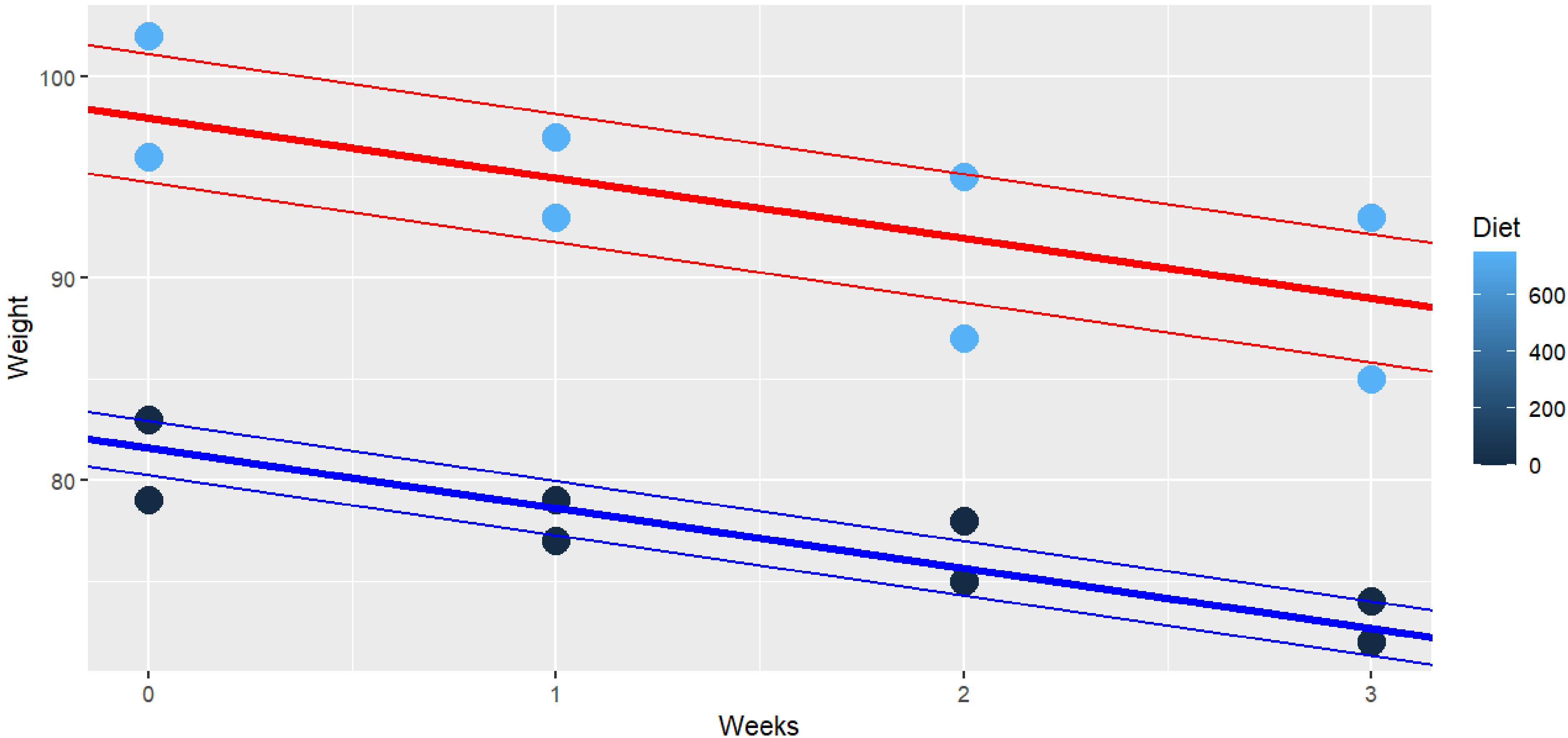
ent

Model with Diet (fixed slope)

Random intercept + Diet

```
# random intercept, fixed slope
lmer0 <- lmer(Weight~Weeks+(1|Person), data=df0)
# random intercept, random slope slope
lmer1 <- lmer(Weight~Weeks+(1+Weeks|Person), data=df0)
# fixed intercept, random slope
lmer2 <- lmer(Weight~Weeks+(0+Weeks|Person), data=df0)
# random intercept + Diet, fixed slope
lmer4 <- lmer(Weight~Weeks+Diet+(1|Person), data=df0)
```

Random Effect is thin line from Thick Fixed Effect line

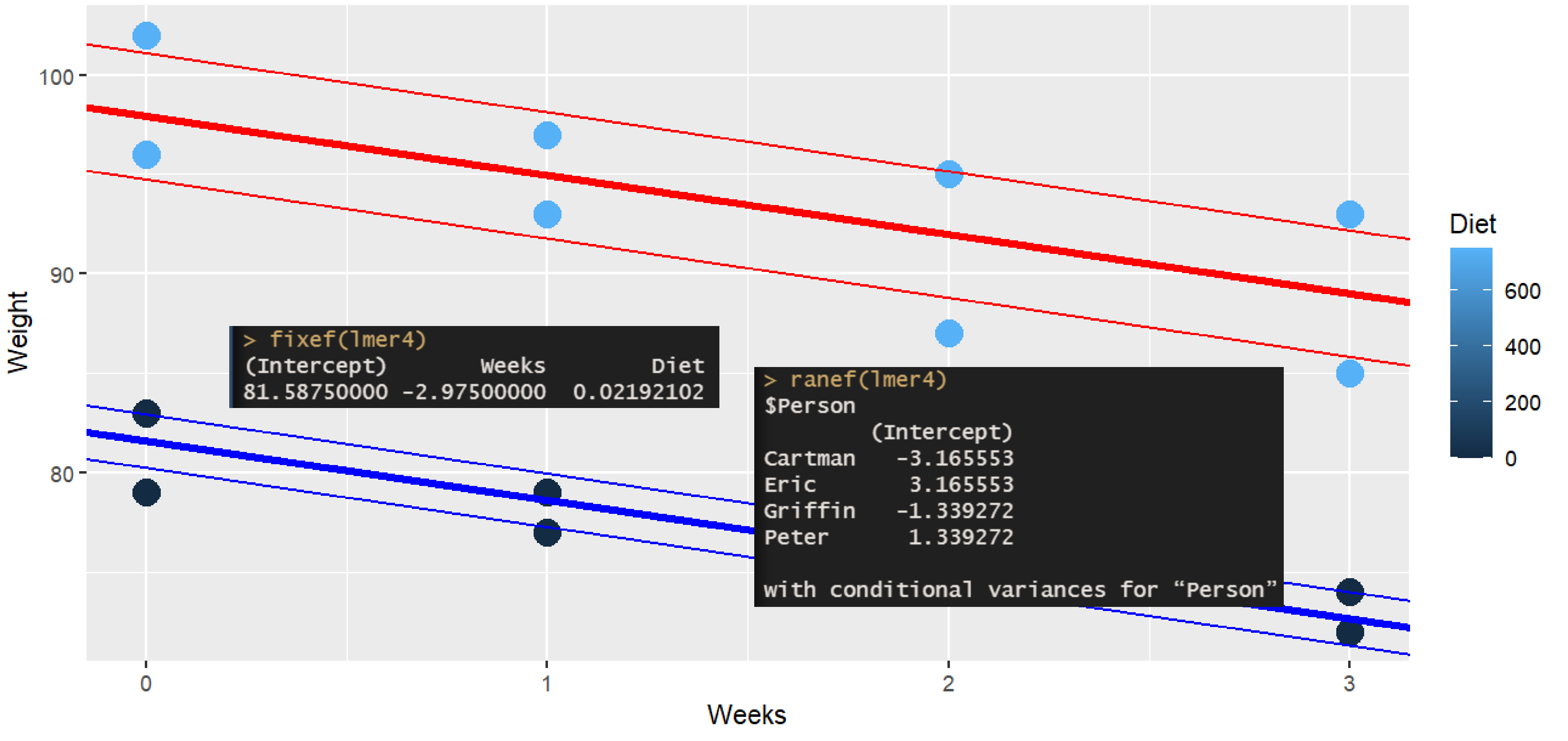


Model with Diet (fixed slope)

Random intercept + Diet

```r
# random intercept, fixed slope
lmer0 <- lmer(Weight~Weeks+(1|Person), data=df0)
# random intercept, random slope slope
lmer1 <- lmer(Weight~Weeks+(1+Weeks|Person), data=df0)
# fixed intercept, random slope
lmer2 <- lmer(Weight~Weeks+(0+Weeks|Person), data=df0)
# random intercept + Diet, fixed slope
lmer4 <- lmer(Weight~Weeks+Diet+(1|Person), data=df0)
```

Random Effect is thin line from Thick Fixed Effect line




```
> fixef(lmer4)
(Intercept)       Weeks        Diet
81.58750000 -2.97500000  0.02192102
```

```
> ranef(lmer4)
$Person
        (Intercept)
Cartman   -3.165553
Eric       3.165553
Griffin   -1.339272
Peter      1.339272

with conditional variances for "Person"
```

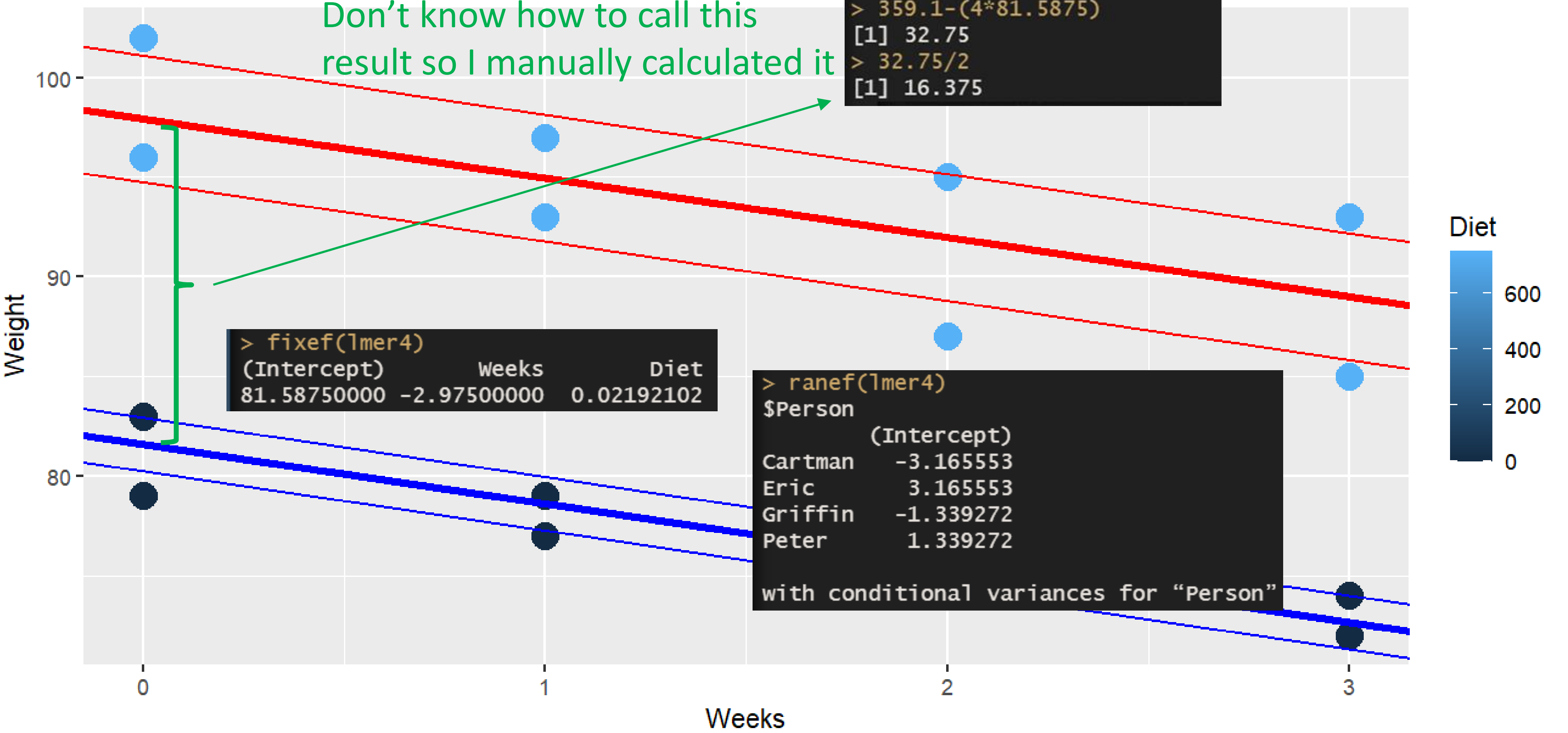
Model with Diet (fixed slope)

Random intercept + Diet

```r
# random intercept, fixed slope
lmer0 <- lmer(Weight~Weeks+(1|Person), data=df0)
# random intercept, random slope slope
lmer1 <- lmer(Weight~Weeks+(1+Weeks|Person), data=df0)
# fixed intercept, random slope
lmer2 <- lmer(Weight~Weeks+(0+Weeks|Person), data=df0)
# random intercept + Diet, fixed slope
lmer4 <- lmer(Weight~Weeks+Diet+(1|Person), data=df0)
```

```
> coef(lmer0)$Person[1]
        (Intercept)
Cartman    94.69614
Eric      101.17461
Griffin    80.24418
Peter      82.98507
> sum(coef(lmer0)$Person[1])
[1] 359.1
> 359.1-(4*81.5875)
[1] 32.75
> 32.75/2
[1] 16.375
```

Don't know how to call this result so I manually calculated it




```
> fixef(lmer4)
(Intercept)       Weeks        Diet
81.58750000 -2.97500000  0.02192102
```

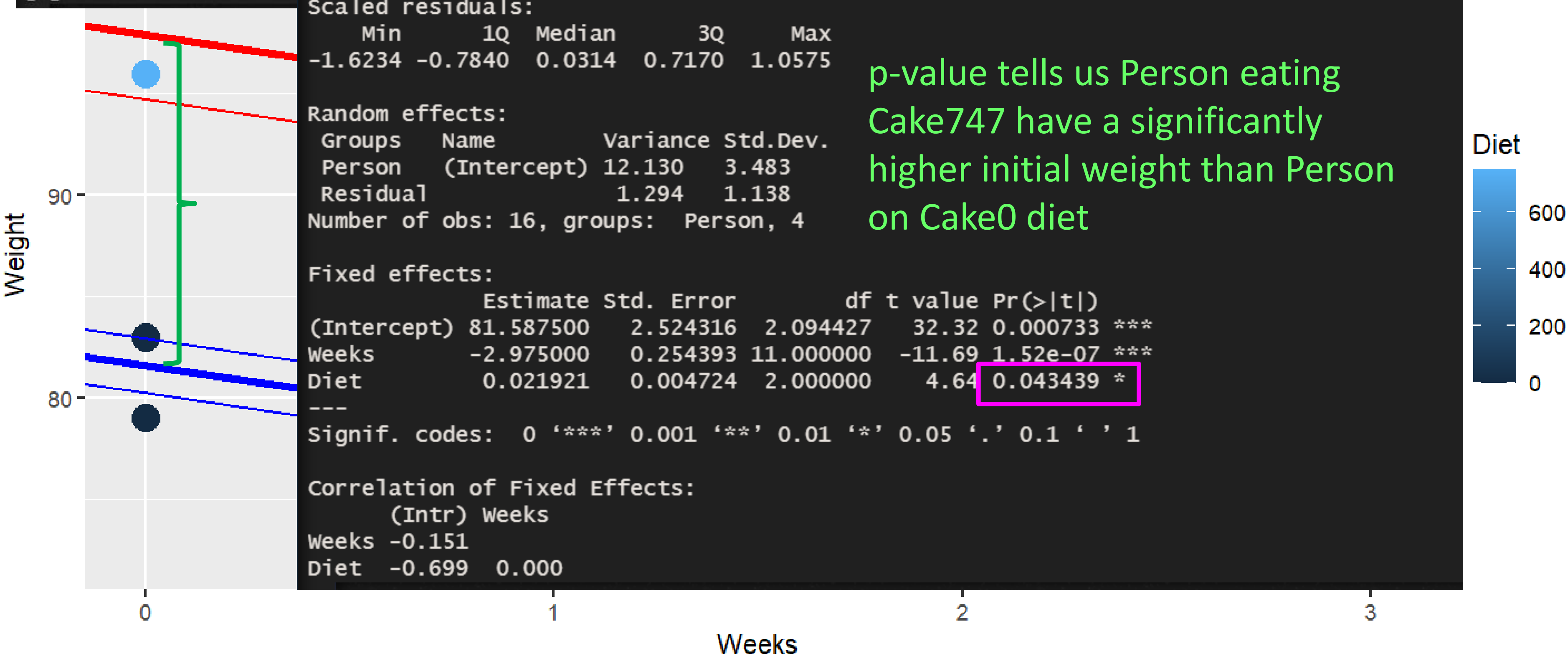
```
> ranef(lmer4)
$Person
        (Intercept)
Cartman   -3.165553
Eric       3.165553
Griffin   -1.339272
Peter      1.339272

with conditional variances for "Person"
```

Random intercept + Diet

```
> coef(lmer0)$Person[1]
        (Intercept)
Cartman    94.69614
Eric      101.17461
Griffin    80.24418
Peter      82.98507
> sum(coef(lmer0)$Per
[1] 359.1
> 359.1-(4*81.5875)
[1] 32.75
> 32.75/2
[1] 16.375
```

```
> summary(lmer4)
Linear mixed model fit by REML. t-tests use Satterthwaite's method ['lmerModLmerTest']
Formula: Weight ~ Weeks + Diet + (1 | Person)
   Data: df0

REML criterion at convergence: 67.9

Scaled residuals:
    Min      1Q  Median      3Q     Max
-1.6234 -0.7840  0.0314  0.7170  1.0575

Random effects:
 Groups   Name        Variance Std.Dev.
 Person   (Intercept) 12.130   3.483
 Residual              1.294   1.138
Number of obs: 16, groups:  Person, 4

Fixed effects:
             Estimate Std. Error        df t value Pr(>|t|)
(Intercept) 81.587500   2.524316  2.094427   32.32 0.000733 ***
Weeks       -2.975000   0.254393 11.000000  -11.69 1.52e-07 ***
Diet         0.021921   0.004724  2.000000    4.64 0.043439 *
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Correlation of Fixed Effects:
      (Intr) Weeks
Weeks -0.151
Diet  -0.699  0.000
```

p-value tells us Person eating Cake747 have a significantly higher initial weight than Person on Cake0 diet



Model with Diet (Random slope)

Is there a significant difference in the Weight loss per Week between the two Diets?

```
# random intercept, fixed slope
lmer0 <- lmer(Weight~Weeks+(1|Person), data=df0)
# random intercept, random slope slope
lmer1 <- lmer(Weight~Weeks+(1+Weeks|Person), data=df0)
# fixed intercept, random slope
lmer2 <- lmer(Weight~Weeks+(0+Weeks|Person), data=df0)
# random intercept + Diet, fixed slope
lmer4 <- lmer(Weight~Weeks+Diet+(1|Person), data=df0)
# random intercept * Diet interaction, fixed slope
lmer5 <- lmer(Weight~Weeks*Diet+(1|Person), data=df0)
```

Weeks*Diet interaction now allow Diet747 to have different slope from Diet0

Model with Diet (Random slope)

Weeks*Diet interaction

Diet0 has slope of -2.55

Diet747 has slope -2.55 – 0.707 = -3.257

p-value of the interaction is not significant. We conclude that the slopes of Diet747 and Diet0 are not significantly different

```
> summary(lmer4)
Linear mixed model fit by REML. t-tests use Satterthwaite's meth
Formula: Weight ~ Weeks + Diet + (1 | Person)
   Data: df0

REML criterion at convergence: 67.9

Scaled residuals:
    Min      1Q  Median      3Q     Max 
-1.6234 -0.7840  0.0314  0.7170  1.0575 

Random effects:
 Groups   Name        Variance Std.Dev.
 Person   (Intercept) 12.130   3.483   
 Residual              1.294   1.138   
Number of obs: 16, groups:  Person, 4

Fixed effects:
             Estimate Std. Error        df t value Pr(>|t|)    
(Intercept) 81.587500   2.524316  2.094427   32.32 0.000733 ***
Weeks       -2.975000   0.254393 11.000000  -11.69 1.52e-07 ***
Diet         0.021921   0.004724  2.000000    4.64 0.043439 *  
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Correlation of Fixed Effects:
      (Intr) Weeks 
Weeks -0.151       
Diet  -0.699  0.000
```

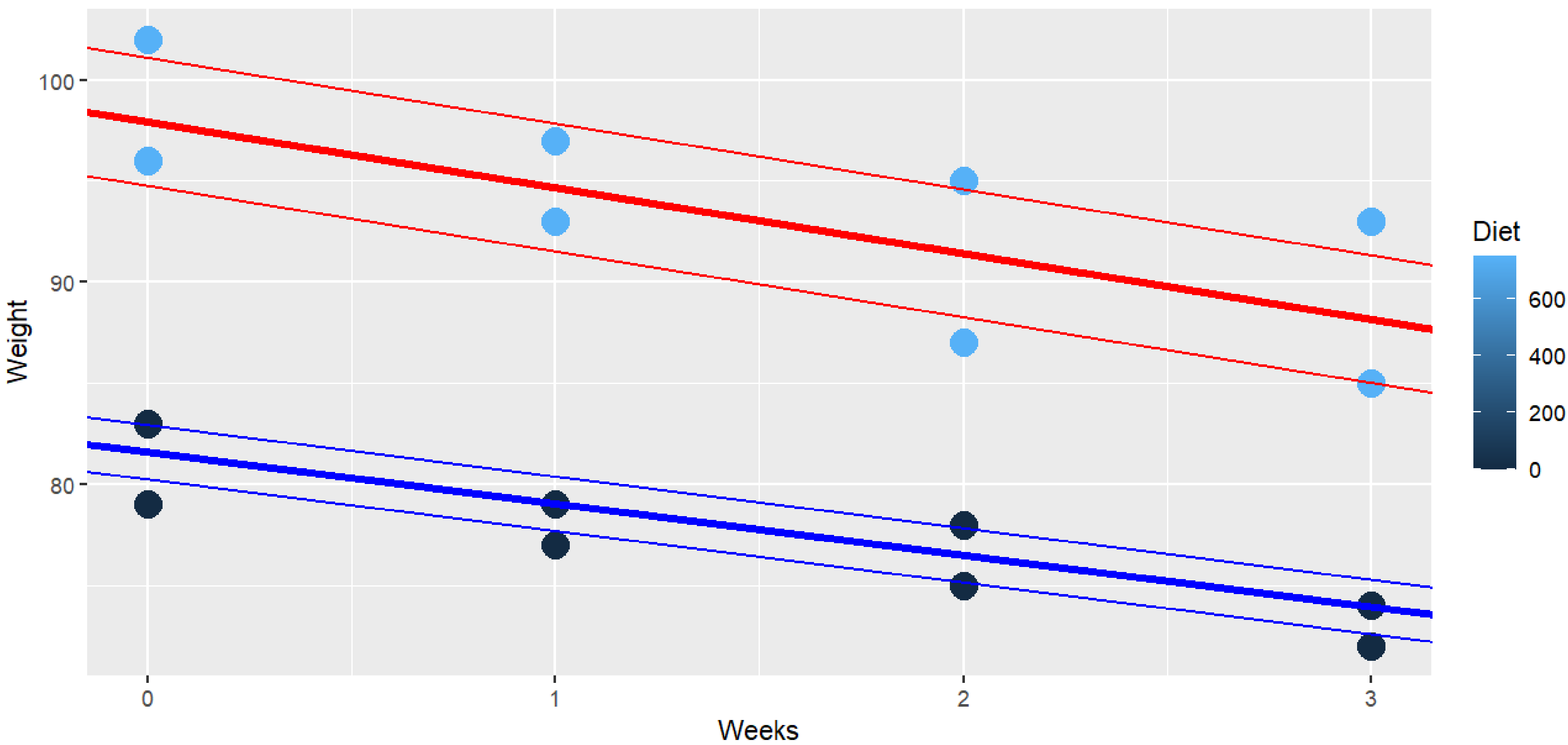
```
> summary(lmer5)
Linear mixed model fit by REML. t-tests use Satterthwaite's metho
Formula: Weight ~ Weeks * Diet + (1 | Person)
   Data: df0

REML criterion at convergence: 77.7

Scaled residuals:
     Min       1Q   Median       3Q      Max 
-1.57097 -0.59631  0.03153  0.59333  1.37695 

Random effects:
 Groups   Name        Variance Std.Dev.
 Person   (Intercept) 12.188   3.491   
 Residual              1.062   1.031   
Number of obs: 16, groups:  Person, 4

Fixed effects:
              Estimate Std. Error         df t value Pr(>|t|)    
(Intercept) 80.9500000  2.5427594  2.1558884  31.835 0.000638 ***
Weeks       -2.5500000  0.3259601 10.0000000  -7.823 1.43e-05 ***
Diet         0.0236278  0.0048139  2.1558884   4.908 0.033556 *  
Weeks:Diet  -0.0011379  0.0006171 10.0000000  -1.844 0.094989 .  
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Correlation of Fixed Effects:
           (Intr) Weeks  Diet  
Weeks      -0.192              
Diet       -0.707  0.136       
Weeks:Diet  0.136 -0.707 -0.192
```

Model with Diet (Random slope)

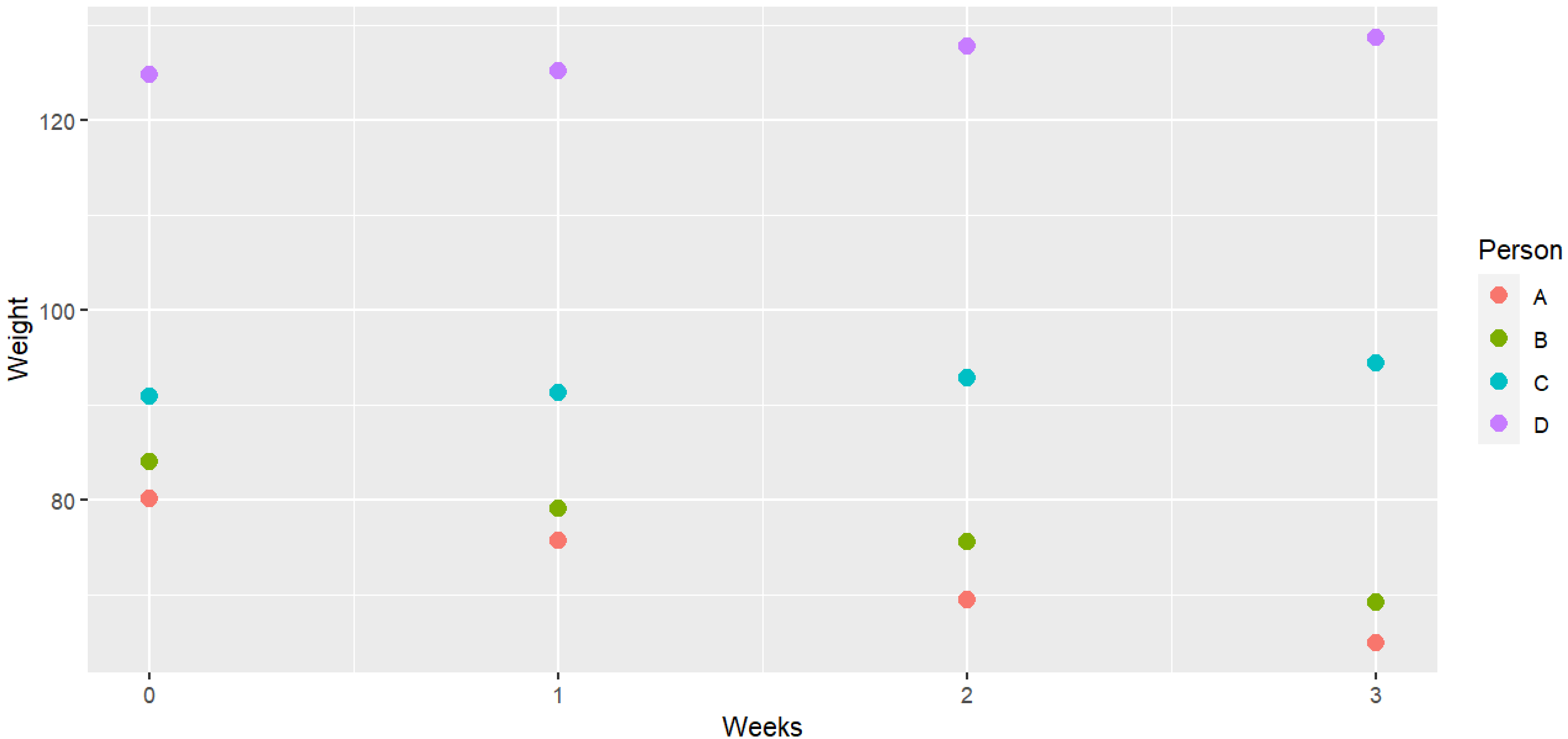
Weeks*Diet interaction

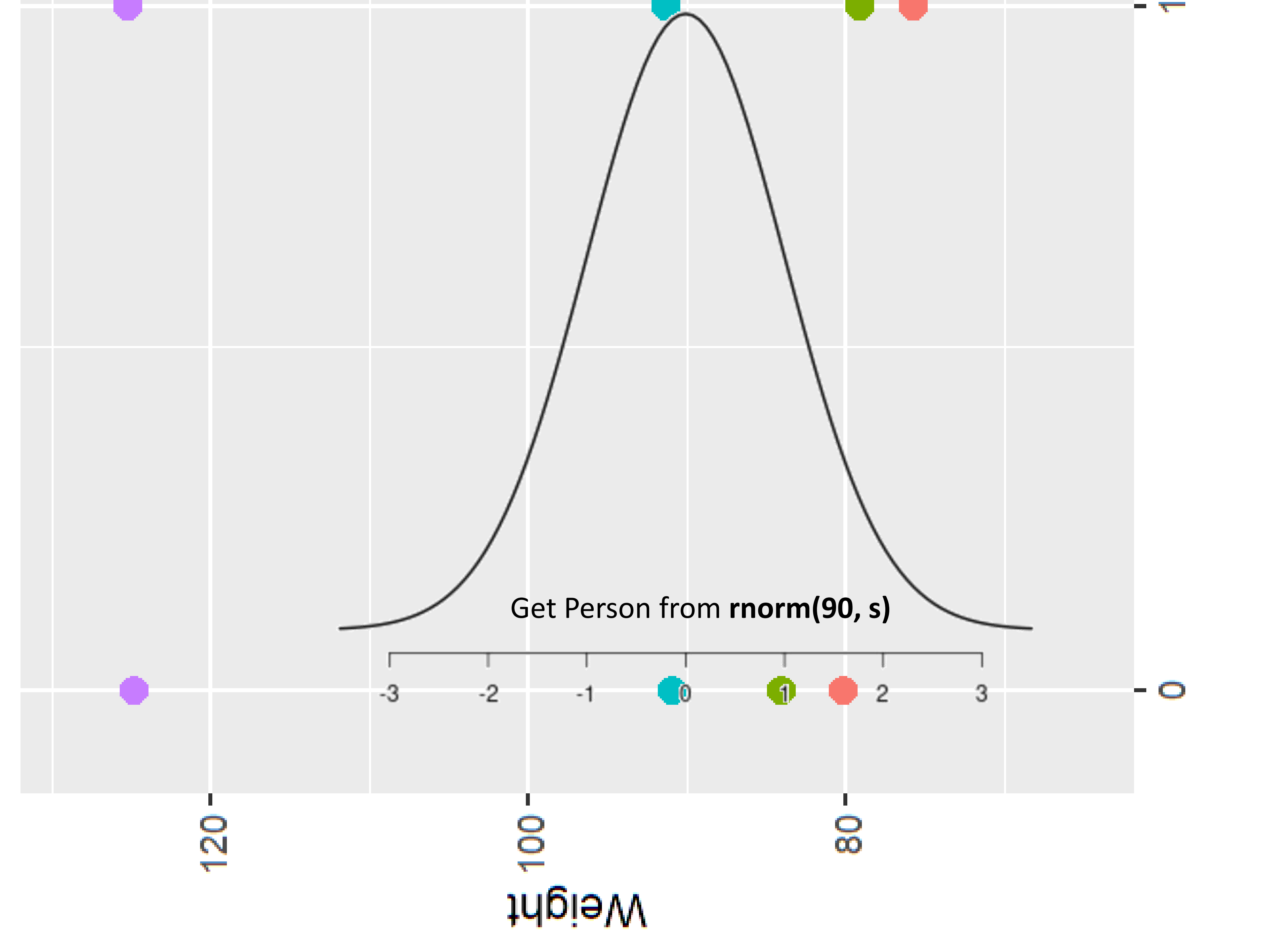
There is no significant difference between average weight loss for Person on Diet with Cake747 and Person on Diet with Cake0

Let’s look at a case where there is a significant difference.



Generate Synthetic Data





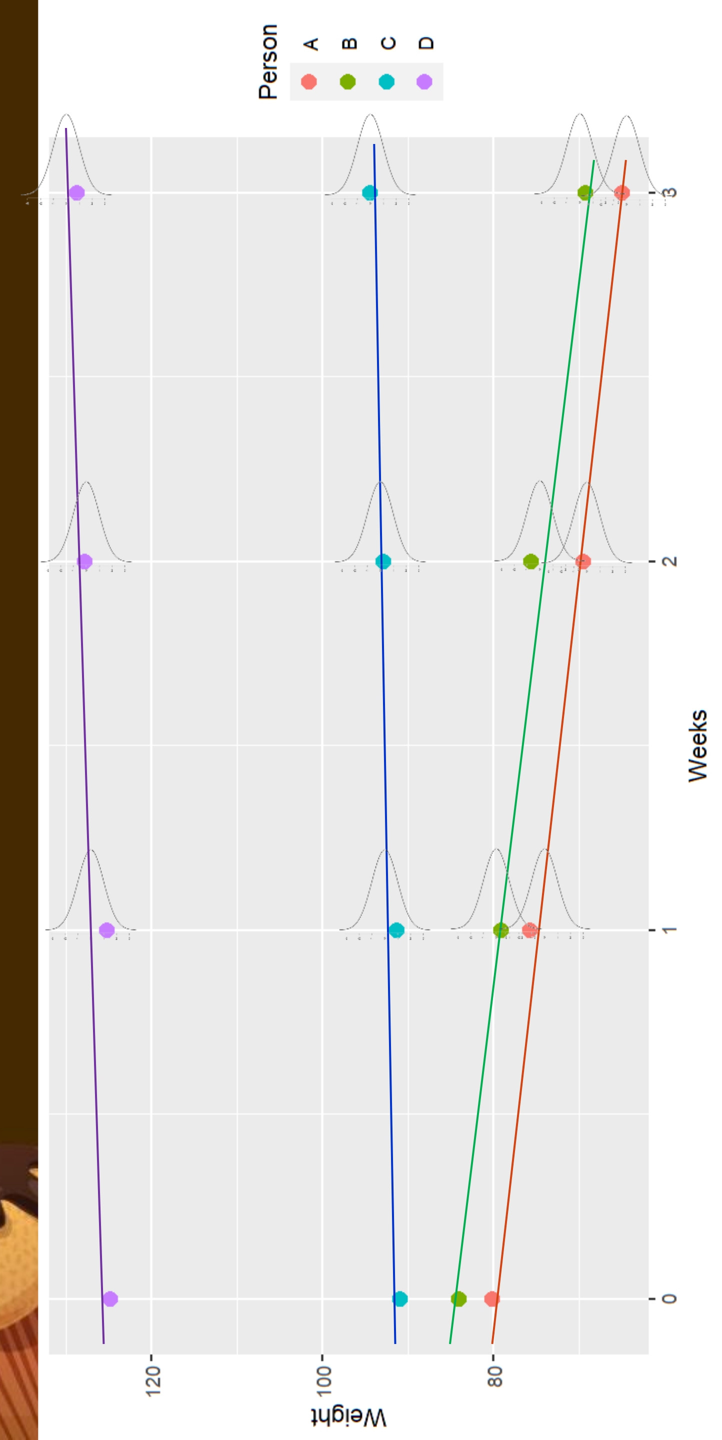
Get Person from **rnorm(90, s)**
-3 -2 -1 0 1 2 3
0
1
120
100
80
Weight

Get Person_i from **rnorm(90, s)**

Calculate predicted $\text{Weight}_{i,j}$ at Week_j

Add Gaussian noise to knock predicted weight randomly off its line

$\text{Weight}_{i,j}$ + rnorm(0, s')



Person{C, D} given Cake0 was generated with flat slope.

Person{A,B} given Cake747 was generated with -5 slope (Weeks)

rnorm(mean=90, sd=15) was used, which SLR got wrong due to 15^2 variation but 2FMM predicted correctly because our two Cake0 happens to be generated next to mean. 2FMM averages Weight of Week0 within each Diet group

We expect simple linear regression (on left) to be insignificant due to high variance 15^2 kg

We expect Two-Factor Mixed Model (on right) to have significant Weeks:Diet interaction

If you've understood this slide, you've understood the entire presentation

```
Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)   97.791      3.203  30.527 3.28e-14 ***
Weeks         -2.753      1.712  -1.608     0.13
```

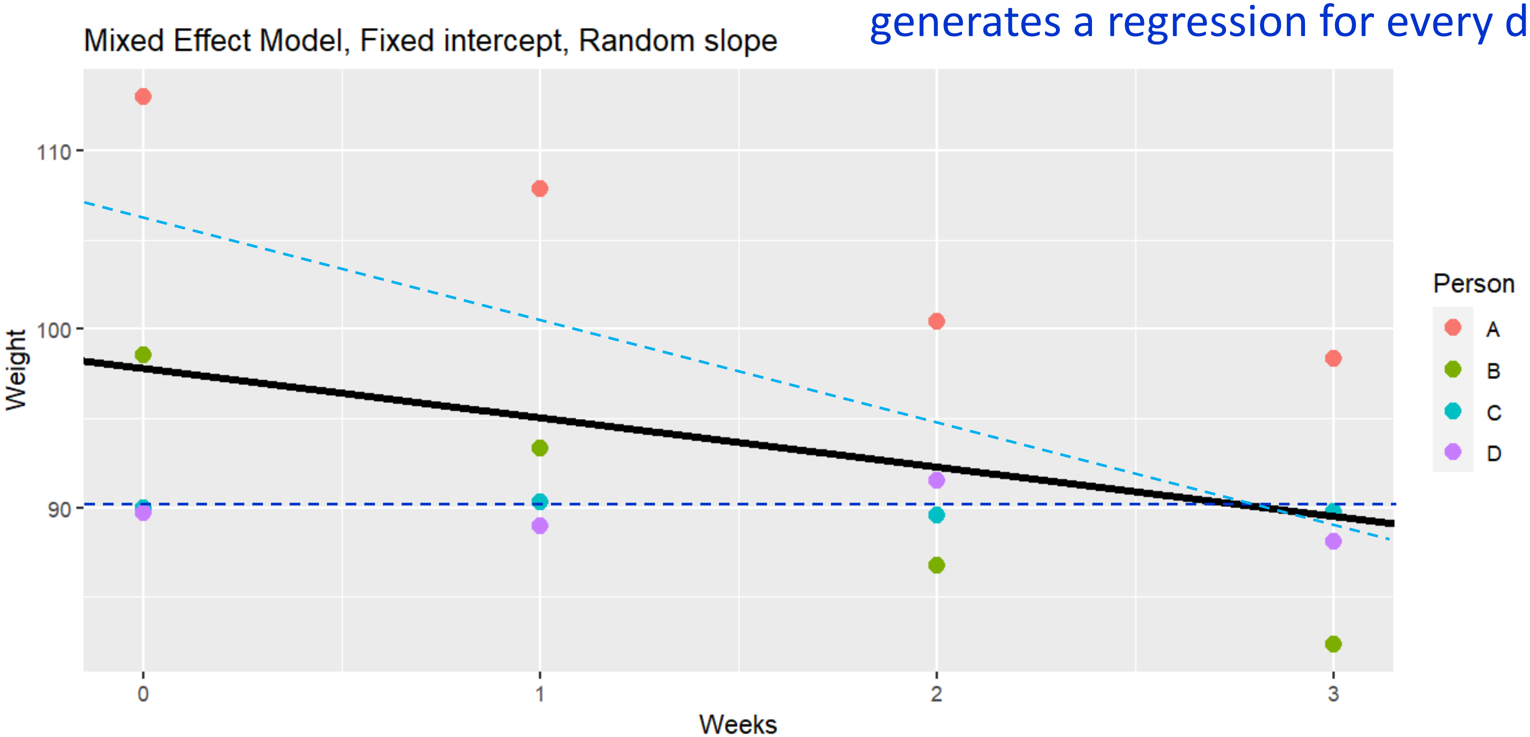
```
synth0.lm <- lm(Weight ~ Weeks, synth0)
summary(synth0.lm)
```

generates an overall regression

```
Fixed effects:
              Estimate Std. Error         df t value Pr(>|t|)
(Intercept) 90.0146054  5.2137050  2.0427617  17.265  0.00304 **
Weeks       -0.1799600  0.3567269 10.0000000  -0.504  0.62486
Diet         0.0208194  0.0098705  2.0427617   2.109  0.16674
Weeks:Diet  -0.0068881  0.0006754 10.0000000 -10.199 1.33e-06 ***
```

```
lmer5 <- lmer(Weight~Weeks*Diet+(1|Person), data=synth0)
summary(lmer5)
```

generates a regression for every diet



Questions?

```r
synthetic <- data.frame(
  Person = LETTERS[1:4],
  Diet = rep(c(747, 0), each=2),
  Week0 = rnorm(4, mean=90, sd=15),
  Week1 = rep(0, 4),
  Week2 = rep(0, 4),
  Week3 = rep(0, 4)
)

# Cake747
b1 <- -5
sd1 <- 1
for(i in 1:2) {
  a <- synthetic$Week0[i]+b1+rnorm(1, mean=0, sd=sd1)
  b <- synthetic$Week0[i]+(2*b1)+rnorm(1, mean=0, sd=sd1)
  c <- synthetic$Week0[i]+(3*b1)+rnorm(1, mean=0, sd=sd1)
  synthetic$Week1[i]=a
  synthetic$Week2[i]=b
  synthetic$Week3[i]=c
}
# Cake0
b2 <- 0
sd2 <- 1
for(i in 3:4) {
  a <- synthetic$Week0[i]+b2+rnorm(1, mean=0, sd=sd2)
  b <- synthetic$Week0[i]+(2*b2)+rnorm(1, mean=0, sd=sd2)
  c <- synthetic$Week0[i]+(3*b2)+rnorm(1, mean=0, sd=sd2)
  synthetic$Week1[i]=a
  synthetic$Week2[i]=b
  synthetic$Week3[i]=c
}
```