Multiple Regression

Tim Frasier

Two Large-scale Problems

1. Underfitting (learning too little)
2. Overfitting (learning “too much”)

Two Large-scale Problems

- Don’t include *everything* as a predictor variable
  - Just variables that you have true hypotheses about

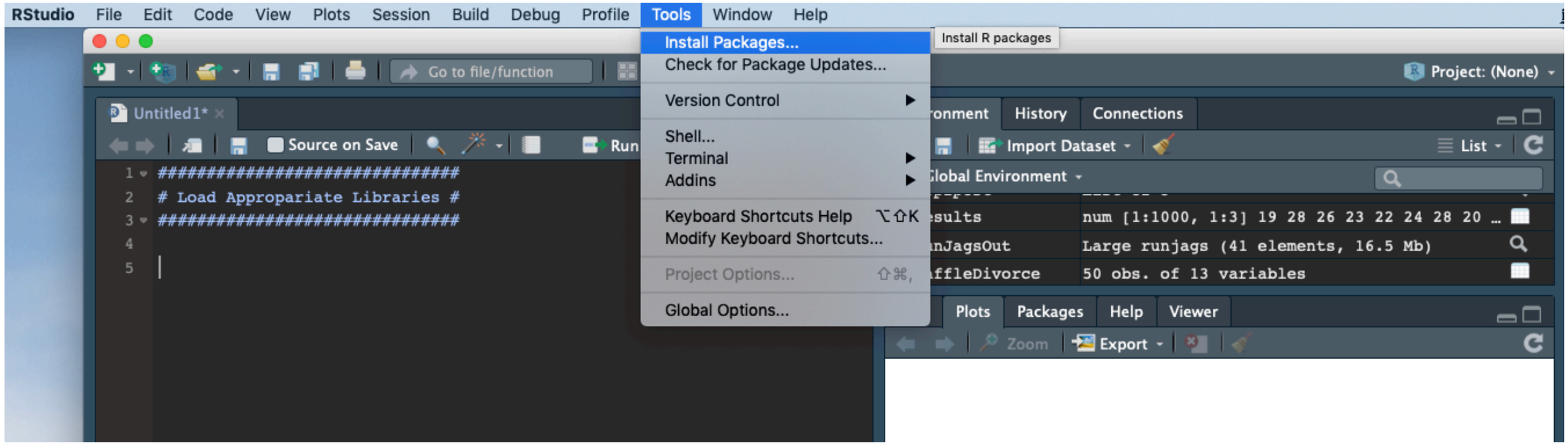
- More variables = more potential problems (interactions)

Principled Workflow

- Will try to teach an approach that will help guide you through this process and avoid pitfalls

1. Plot all pairs of variables (don't take at face value yet!)
   a. Note all relationships
   b. Are any relationships potentially "masking" or interfering?
2. Check correlation among predictor variables
   a. Make note of all correlations > | 0.7 |
   b. Try model with both, then each on their own
3. Build/run multiple models based on notes from above
   a. Use WAIC to compare performance and characteristics of models
   b. Don't necessarily choose one model, but base interpretation on all
4. Interpret parameters based on simulations, not actual estimates

- Load the `loo` package
  - For WAIC calculations



Data

TV stars and coaches charged in college bribery scheme


RAD
A hometown kid on his BMX
against the best in the world.
At Helltrack... the heat is on.
JACK SCHWARTZMAN PRESENTS
TAHAFILM II LTD./ROBERT L. LEVY PRODUCTION
A HAL NEEDHAM FILM "RAD" BILL ALLEN · LORI LOUGHLIN
TALIA SHIRE · RAY WALSTON
BART CONNER AND JACK WESTON
A JAMES DI PASQUALE FILM
CO-PRODUCED BY SAM BERNARD
EXECUTIVE PRODUCER JACK SCHWARTZMAN
SAM BERNARD & GEOFFREY EDWARDS
PRODUCED BY ROBERT L. LEVY
WRITTEN BY HAL NEEDHAM


1

Data

- SAT information from Guber (1999), as provided by Kruschke (2015)
  - "`Guber1999data.csv`"

```
sat = read.table("Guber1999data.csv", header = TRUE, sep = ",")
```

Guber (1999) Journal of Statistics Education 7(2).

Data

guber

50 observations of 8

	State	Spend	StuTeaRat	Salary	PrcntTake	SATV	SATM	SATT
1	Alabama	4.405	17.2	31.144	8	491	538	1029
2	Alaska	8.963	17.6	47.951	47	445	489	934
3	Arizona	4.778	19.3	32.175	27	448	496	944
4	Arkansas	4.459	17.1	28.934	6	482	523	1005
5	California	4.992	24.0	41.078	45	417	485	902
6	Colorado	5.443	18.4	34.571	29	462	518	980
7	Connecticut	8.817	14.4	50.045	81	431	477	908
8	Delaware	7.030	16.6	39.076	68	429	468	897
9	Florida	5.718	19.1	32.588	48	420	469	889
10	Georgia	5.193	16.3	32.291	65	406	448	854
11	Hawaii	6.078	17.9	38.518	57	407	482	889
12	Idaho	4.210	19.1	29.783	15	468	511	979
13	Illinois	6.136	17.3	39.431	13	488	560	1048
14	Indiana	5.826	17.5	36.785	58	415	467	882
15	Iowa	5.483	15.8	31.511	5	516	583	1099
16	Kansas	5.817	15.1	34.652	9	503	557	1060

Data

guber

50 observations of 8

	State	Spend	StuTeaRat	Salary	PrcntTake	SATV	SATM	SATT
1	Alabama	4.405	17.2	31.144	8	491	538	1029
2	Alaska	8.963	17.6	47.951	47	445	489	934
3	Arizona	4.778	19.3	32.175	27	448	496	944
4	Arkansas	4.459	17.1	28.934	6	482	523	1005
5	California	4.992	24.0	41.078	45	417	485	902
6	Colorado					462	518	980
7	Connecticut					431	477	908
8	Delaware					429	468	897
9	Florida					420	469	889
10	Georgia	5.193	16.3	32.291	65	406	448	854
11	Hawaii	6.078	17.9	38.518	57	407	482	889
12	Idaho	4.210	19.1	29.783	15	468	511	979
13	Illinois	6.136	17.3	39.431	13	488	560	1048
14	Indiana	5.826	17.5	36.785	58	415	467	882
15	Iowa	5.483	15.8	31.511	5	516	583	1099
16	Kansas	5.817	15.1	34.652	9	503	557	1060

Self explanatory

Data

guber

50 observations of 8

	State	Spend	StuTeaRat	Salary	PrcntTake	SATV	SATM	SATT
1	Alabama	4.405	17.2	31.144	8	491	538	1029
2	Alaska	8.963	17.6	47.951	47	445	489	934
3	Arizona	4.778	19.3	32.175	27	448	496	944
4	Arkansas	4.459	17.1	28.934	6	482	523	1005
5	California	4.992	24.0	41.078	45	417	485	902
6	Colorado							980
7	Connecticut							908
8	Delaware							897
9	Florida							889
10	Georgia	5.193	16.3	32.291	65	406	448	854
11	Hawaii	6.078	17.9	38.518	57	407	482	889
12	Idaho	4.210	19.1	29.783	15	468	511	979
13	Illinois	6.136	17.3	39.431	13	488	560	1048
14	Indiana	5.826	17.5	36.785	58	415	467	882
15	Iowa	5.483	15.8	31.511	5	516	583	1099
16	Kansas	5.817	15.1	34.652	9	503	557	1060

The average amount the state spends per student (in thousands of dollars).

Data

guber

50 observations of 8

	State	Spend	StuTeaRat	Salary	PrcntTake	SATV	SATM	SATT
1	Alabama	4.405	17.2	31.144	8	491	538	1029
2	Alaska	8.963	17.6	47.951	47	445	489	934
3	Arizona	4.778	19.3	32.175	27	448	496	944
4	Arkansas	4.459	17.1	28.934	6	482	523	1005
5	California	4.992	24.0	41.078	45	417	485	902
6	Colorado							980
7	Connecticut							908
8	Delaware							897
9	Florida							889
10	Georgia	5.193	16.3	32.291	65	406	448	854
11	Hawaii	6.078	17.9	38.518	57	407	482	889
12	Idaho	4.210	19.1	29.783	15	468	511	979
13	Illinois	6.136	17.3	39.431	13	488	560	1048
14	Indiana	5.826	17.5	36.785	58	415	467	882
15	Iowa	5.483	15.8	31.511	5	516	583	1099
16	Kansas	5.817	15.1	34.652	9	503	557	1060

Average student to teacher ratio.

Data

guber

50 observations of 8

	State	Spend	StuTeaRat	Salary	PrcntTake	SATV	SATM	SATT
1	Alabama	4.405	17.2	31.144	8	491	538	1029
2	Alaska	8.963	17.6	47.951	47	445	489	934
3	Arizona	4.778	19.3	32.175	27	448	496	944
4	Arkansas	4.459	17.1	28.934	6	482	523	1005
5	California	4.992	24.0	41.078	45	417	485	902
6	Colorado							980
7	Connecticut							908
8	Delaware							897
9	Florida							889
10	Georgia	5.193	16.3	32.291	65	406	448	854
11	Hawaii	6.078	17.9	38.518	57	407	482	889
12	Idaho	4.210	19.1	29.783	15	468	511	979
13	Illinois	6.136	17.3	39.431	13	488	560	1048
14	Indiana	5.826	17.5	36.785	58	415	467	882
15	Iowa	5.483	15.8	31.511	5	516	583	1099
16	Kansas	5.817	15.1	34.652	9	503	557	1060

Average teacher salary

Data

guber

50 observations of 8

	State	Spend	StuTeaRat	Salary	PrcntTake	SATV	SATM	SATT
1	Alabama	4.405	17.2	31.144	8	491	538	1029
2	Alaska	8.963	17.6	47.951	47	445	489	934
3	Arizona	4.778	19.3	32.175	27	448	496	944
4	Arkansas	4.459	17.1	28.934	6	482	523	1005
5	California	4.992	24.0	41.078	45	417	485	902
6	Colorado							980
7	Connecticut							908
8	Delaware							897
9	Florida							889
10	Georgia	5.193	16.3	32.291	65	406	448	854
11	Hawaii	6.078	17.9	38.518	57	407	482	889
12	Idaho	4.210	19.1	29.783	15	468	511	979
13	Illinois	6.136	17.3	39.431	13	488	560	1048
14	Indiana	5.826	17.5	36.785	58	415	467	882
15	Iowa	5.483	15.8	31.511	5	516	583	1099
16	Kansas	5.817	15.1	34.652	9	503	557	1060

Percent of students who take the SAT.

Data

guber

50 observations of 8

	State	Spend	StuTeaRat	Salary	PrcntTake	SATV	SATM	SATT
1	Alabama	4.405	17.2	31.144	8	491	538	1029
2	Alaska	8.963	17.6	47.951	47	445	489	934
3	Arizona	4.778	19.3	32.175	27	448	496	944
4	Arkansas	4.459	17.1	28.934	6	482	523	1005
5	California	4.992	24.0	41.078	45	417	485	902
6	Colorado							980
7	Connecticut							908
8	Delaware							897
9	Florida							889
10	Georgia	5.193	16.3	32.291	65	406	448	854
11	Hawaii	6.078	17.9	38.518	57	407	482	889
12	Idaho	4.210	19.1	29.783	15	468	511	979
13	Illinois	6.136	17.3	39.431	13	488	560	1048
14	Indiana	5.826	17.5	36.785	58	415	467	882
15	Iowa	5.483	15.8	31.511	5	516	583	1099
16	Kansas	5.817	15.1	34.652	9	503	557	1060

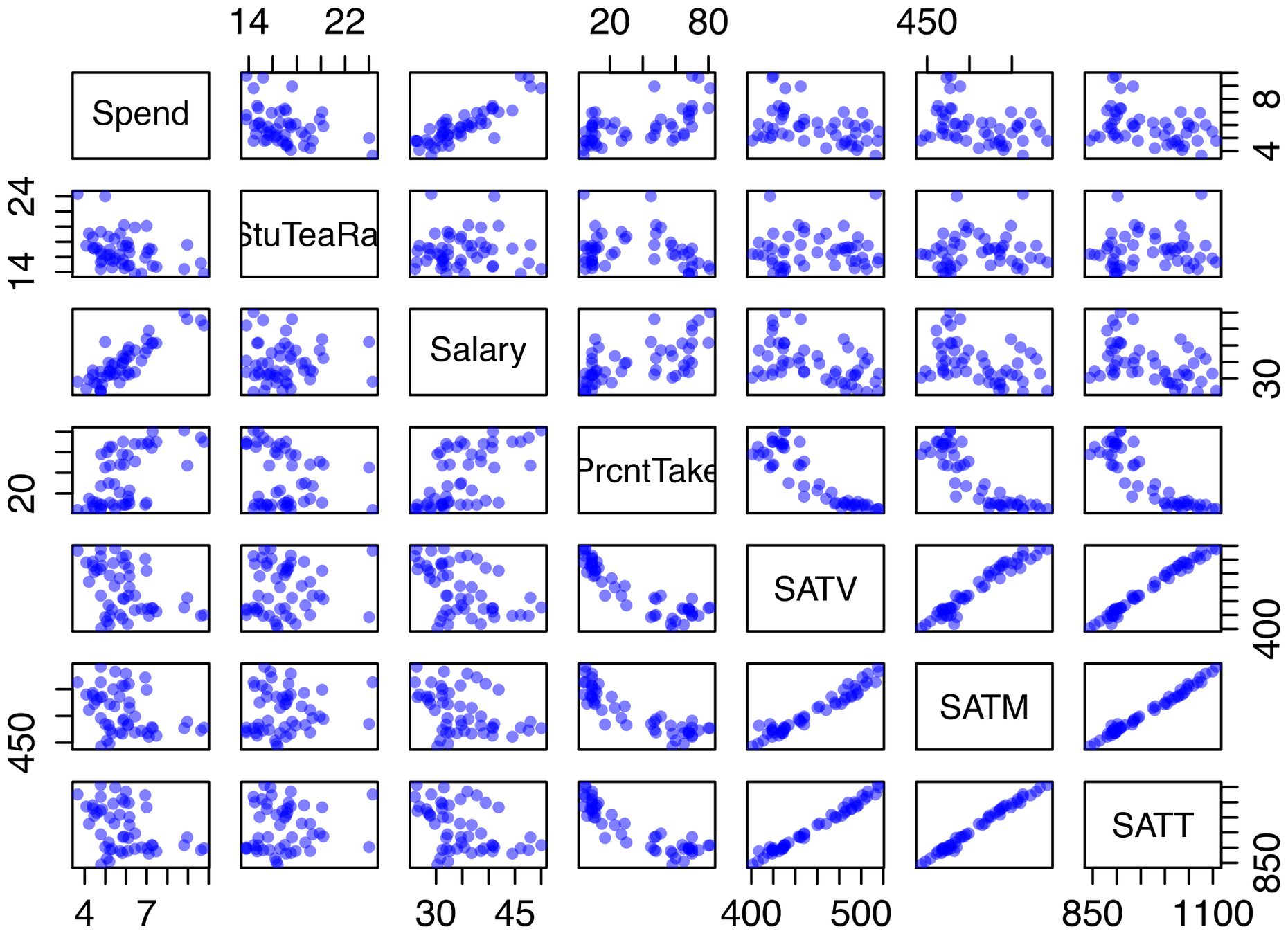
Average verbal, math, and total SAT scores for the state.

Data

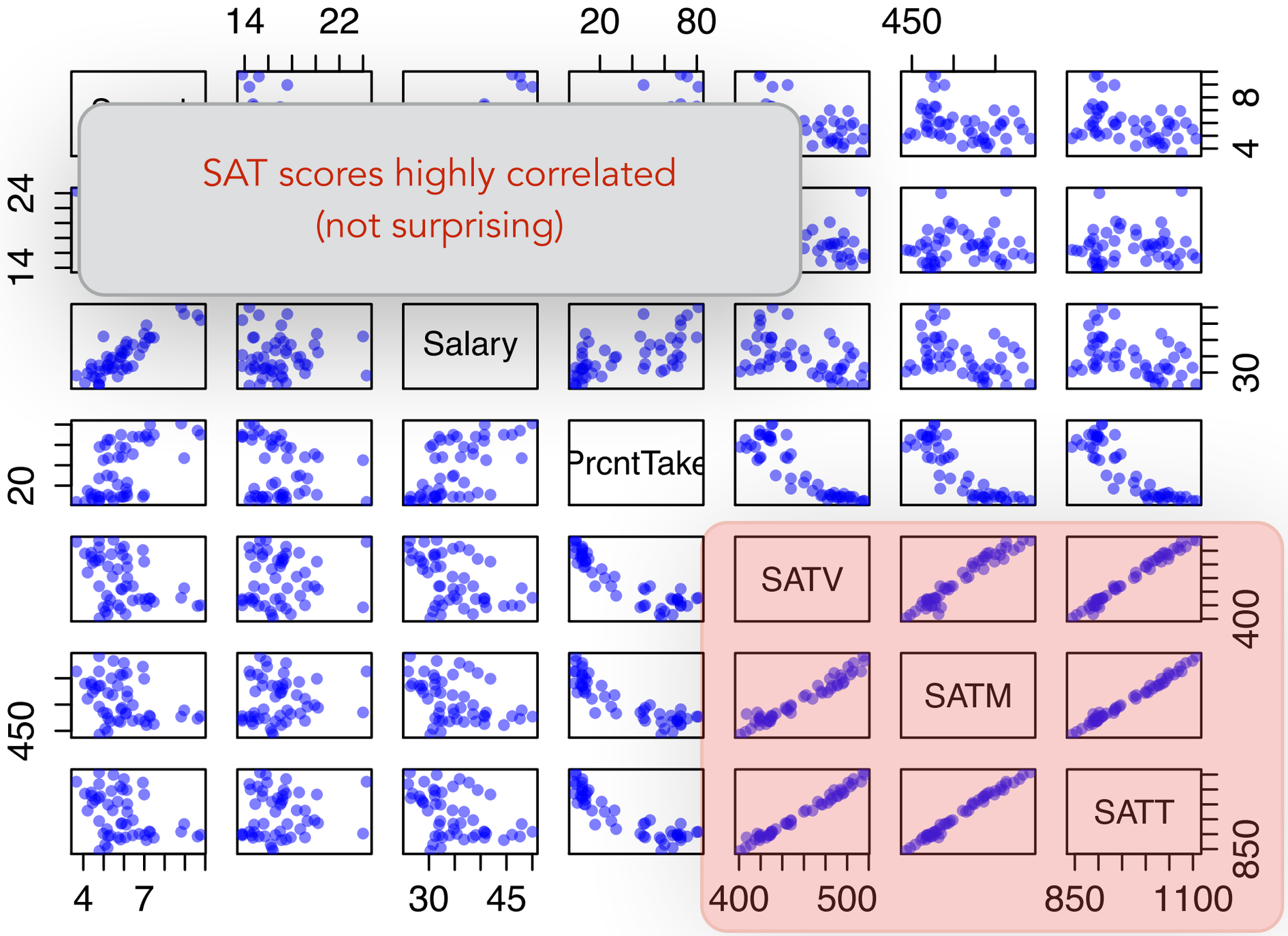
- Plot the data. `pairs` function shows all pairwise comparisons.
  - We'll leave out the `state` field

```
pairs(guber[, 2:8], pch = 16, col = rgb(0, 0, 1, 0.5))
```

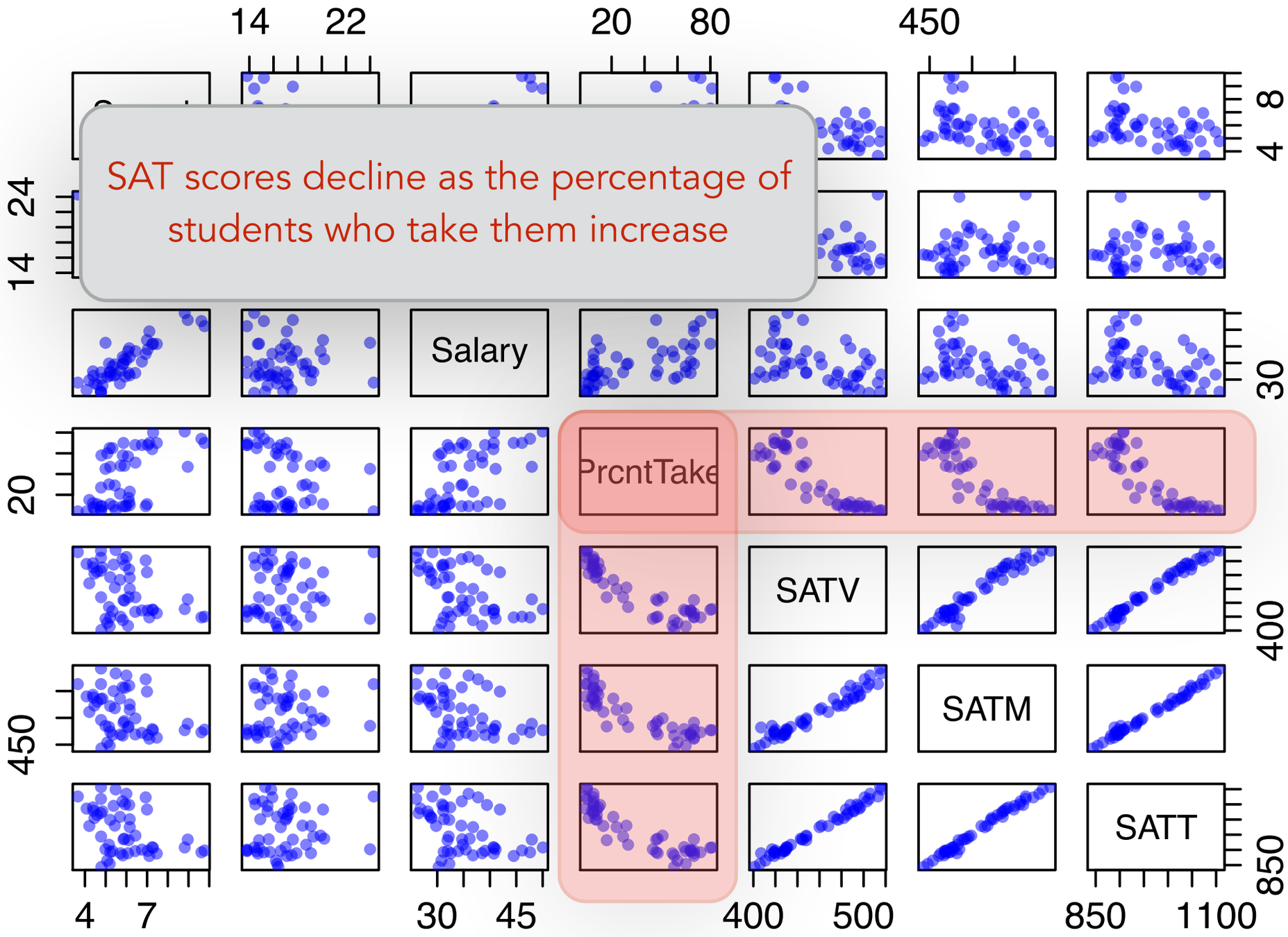
Data


Spend
StuTeaRa
Salary
PrcntTake
SATV
SATM
SATT
14
22
20
80
450
4
8
14
24
20
450
400
850
4
7
30
45
400
500
850
1100

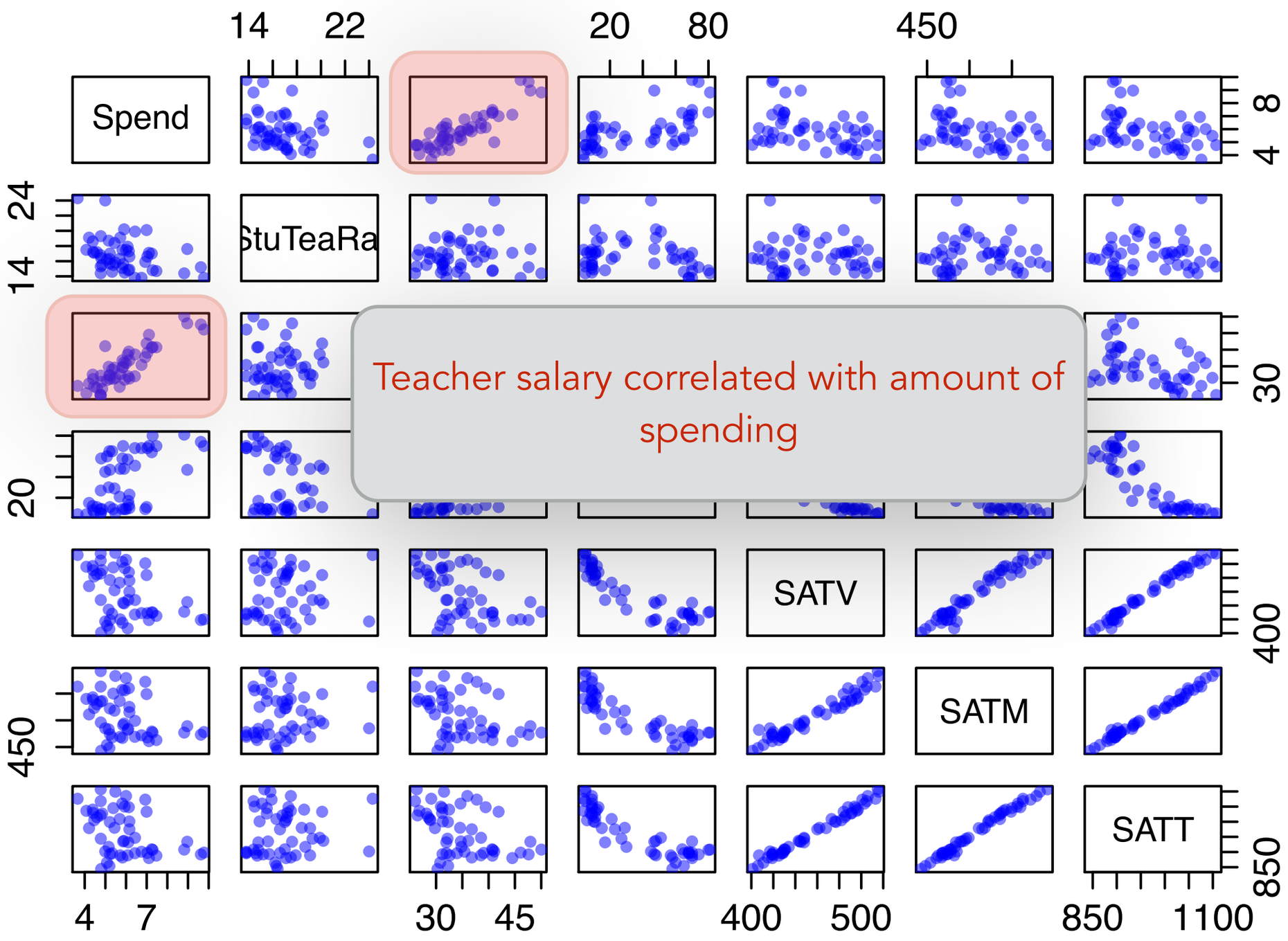
Data


14
22
20
80
450
4
8
24
14
SAT scores highly correlated
(not surprising)
Salary
PrcntTake
20
SATV
400
SATM
450
SATT
850
4
7
30
45
400
500
850
1100

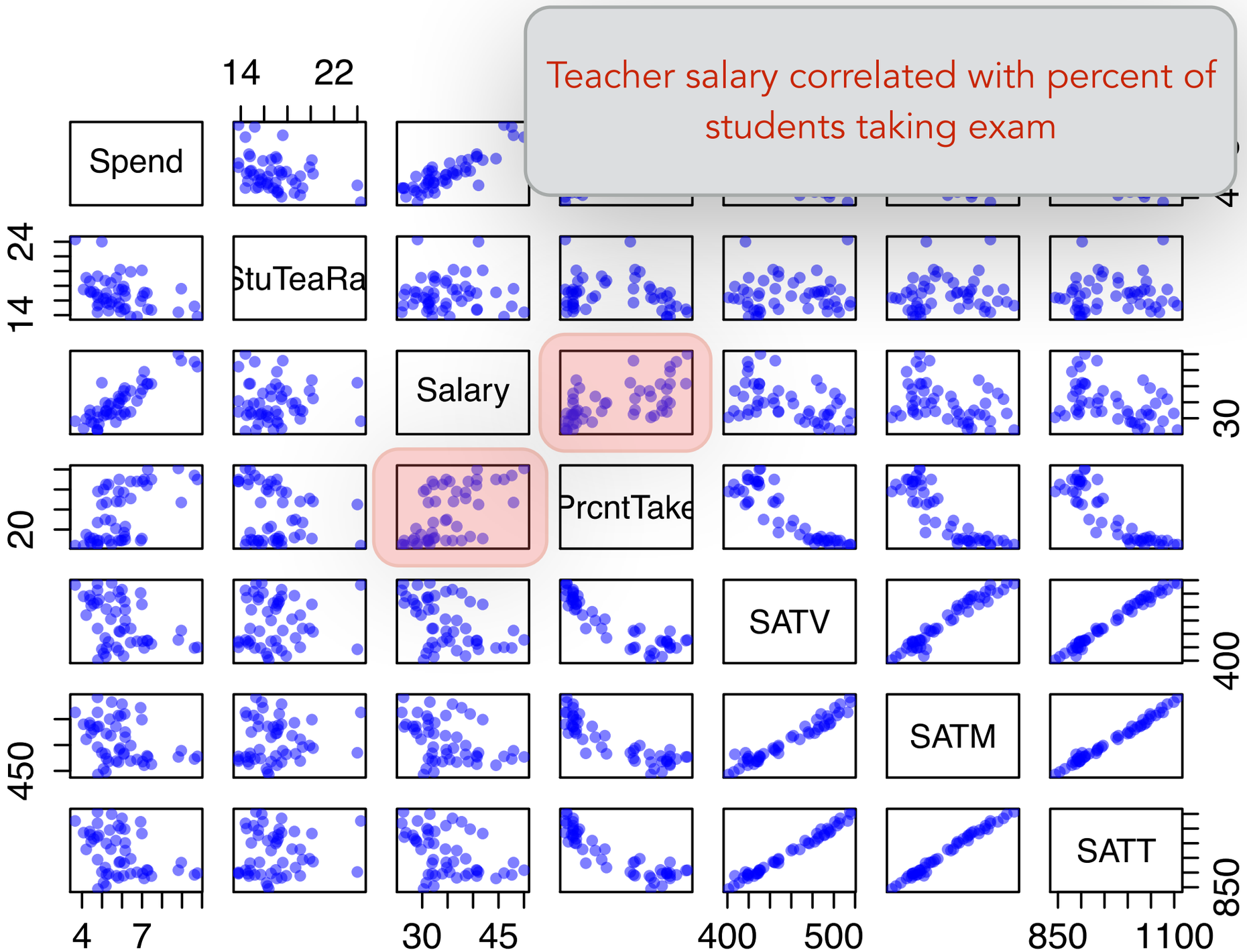
Data


14
22
20
80
450
4
8
24
14
SAT scores decline as the percentage of students who take them increase
Salary
PrcntTake
SATV
SATM
SATT
20
400
450
850
4
7
30
45
400
500
850
1100

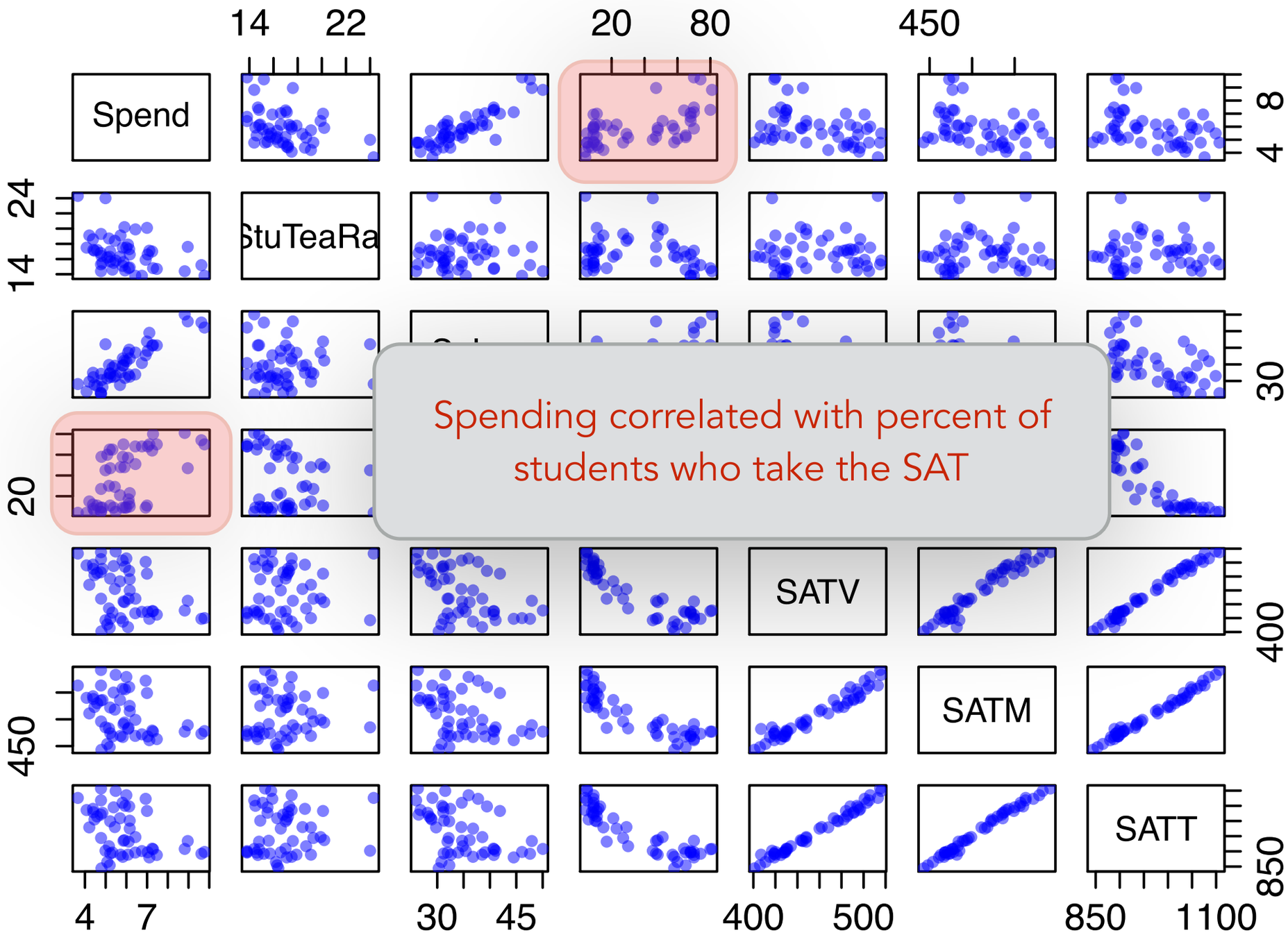
Data


Spend
StuTeaRa
SATV
SATM
SATT
14
22
20
80
450
4
8
24
14
20
450
30
400
850
4
7
30
45
400
500
850
1100
Teacher salary correlated with amount of spending

Data

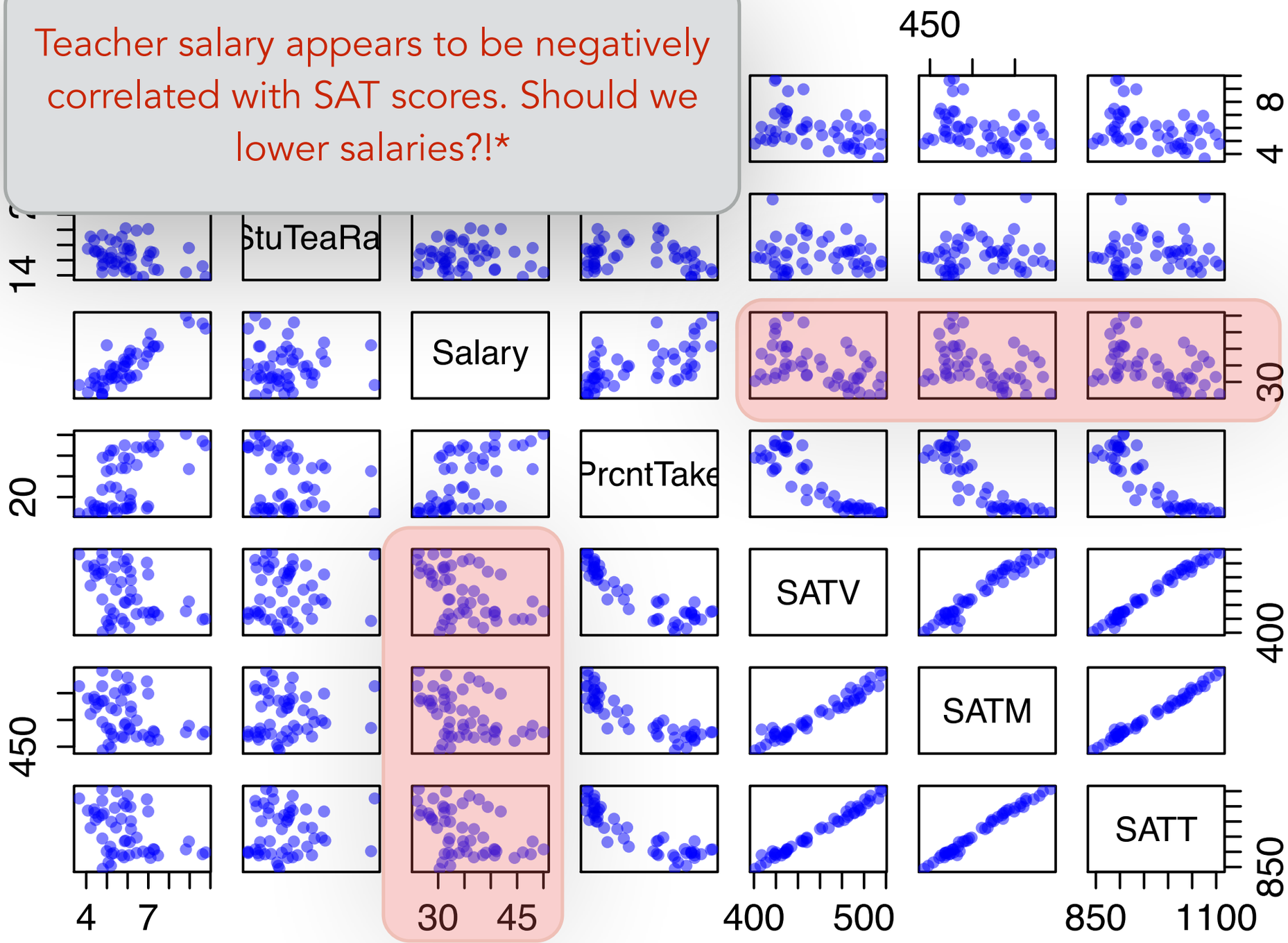

Teacher salary correlated with percent of students taking exam
Spend
StuTeaRa
Salary
PrcntTake
SATV
SATM
SATT
14
22
24
14
20
4
7
30
45
400
500
850
1100
450
400
850

Data


Spend
StuTeaRa
SATV
SATM
SATT
14
22
20
80
450
4
8
24
14
20
30
450
400
850
4
7
30
45
400
500
850
1100
Spending correlated with percent of students who take the SAT

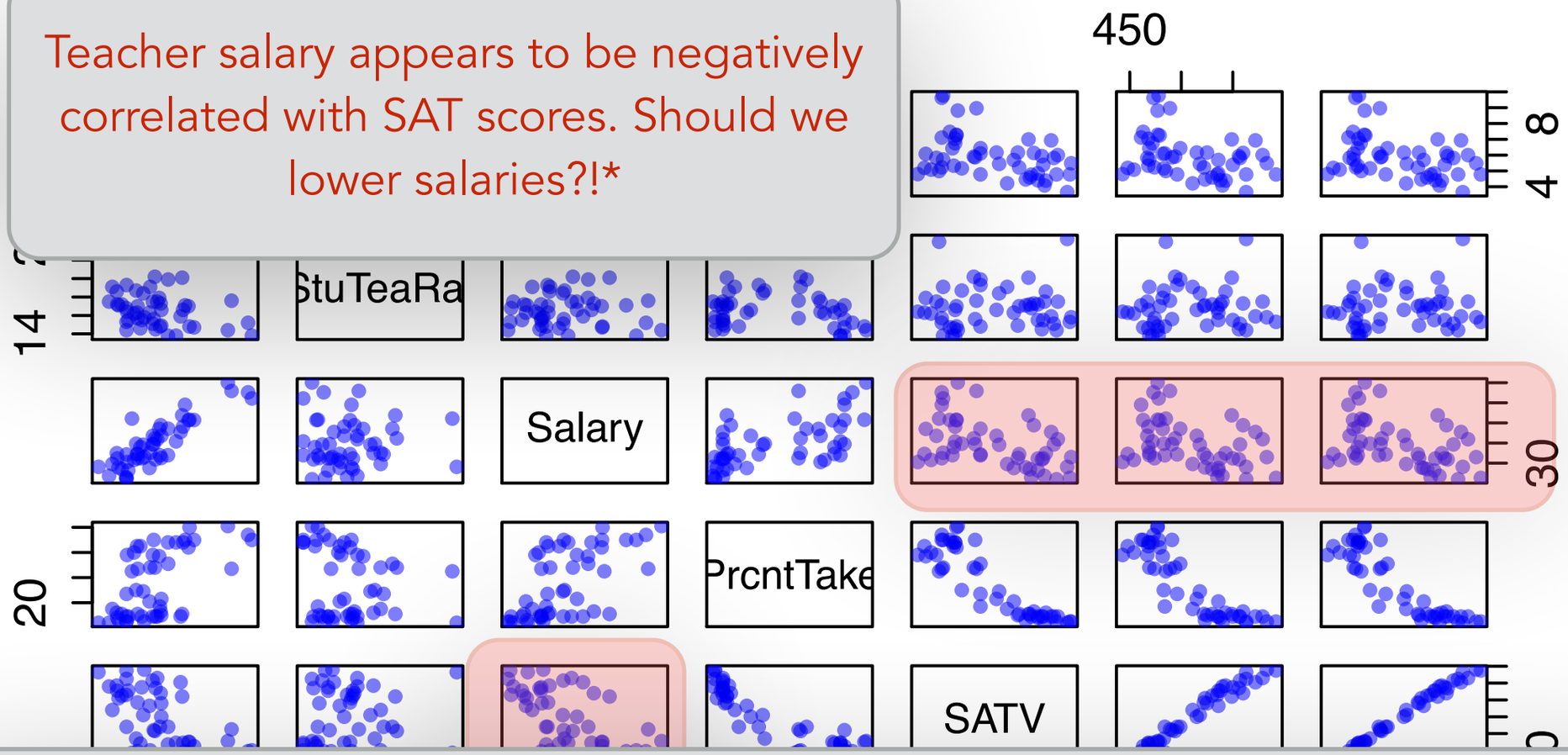
Data

Teacher salary appears to be negatively correlated with SAT scores. Should we lower salaries?!*

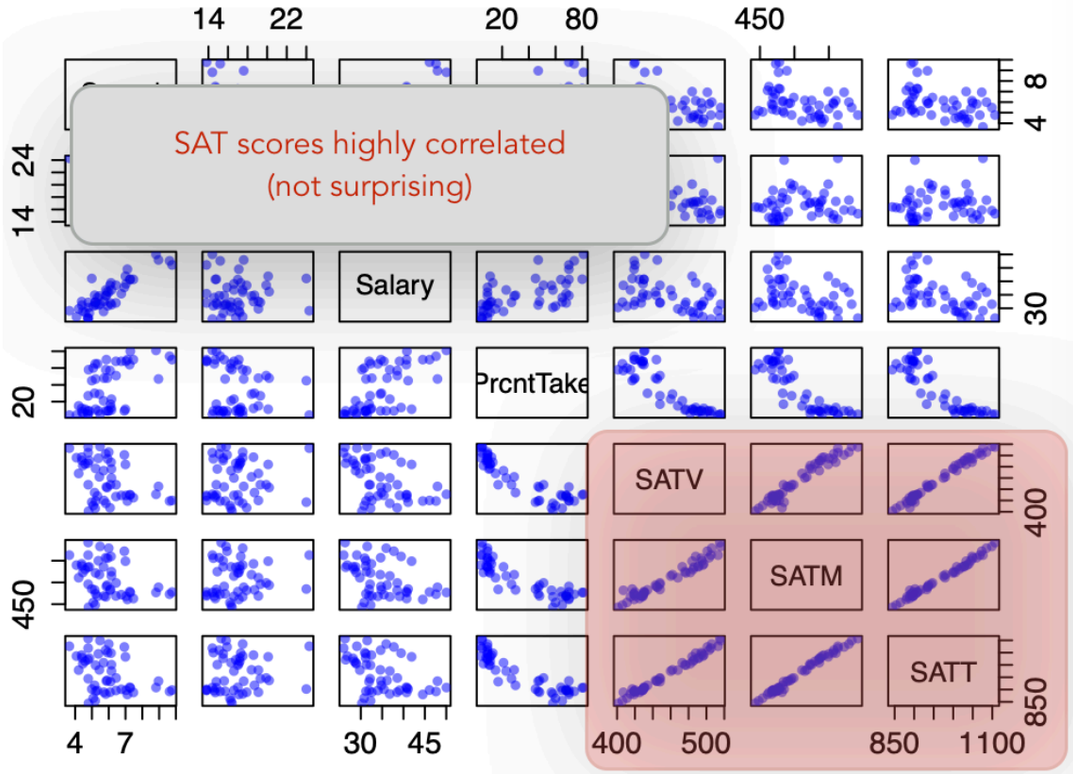


Data

Teacher salary appears to be negatively correlated with SAT scores. Should we lower salaries?!*



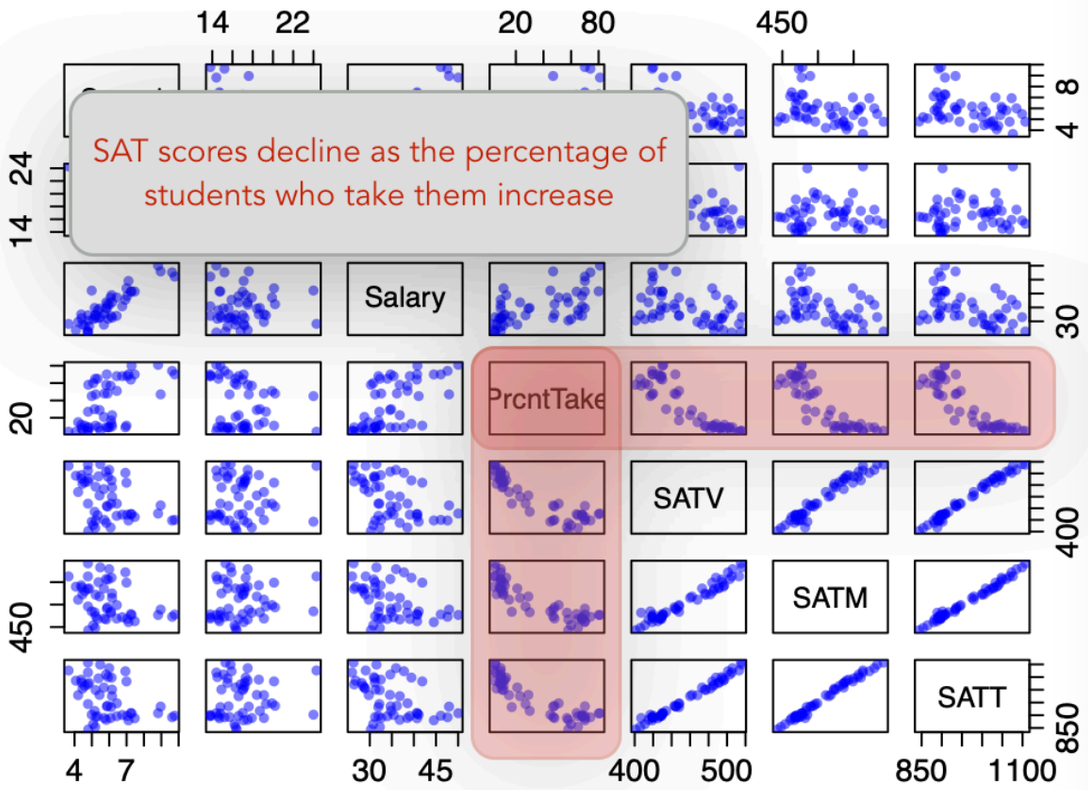
* As we'll see, this is an artefact of correlation among predictor variables. And the true relationship is positive. This highlights one strength of multiple linear regression (where multiple variables can be considered simultaneously), as well as some of the dangers of taking too simple of an approach/interpretation.



SAT Score Analysis

1: Initial plots

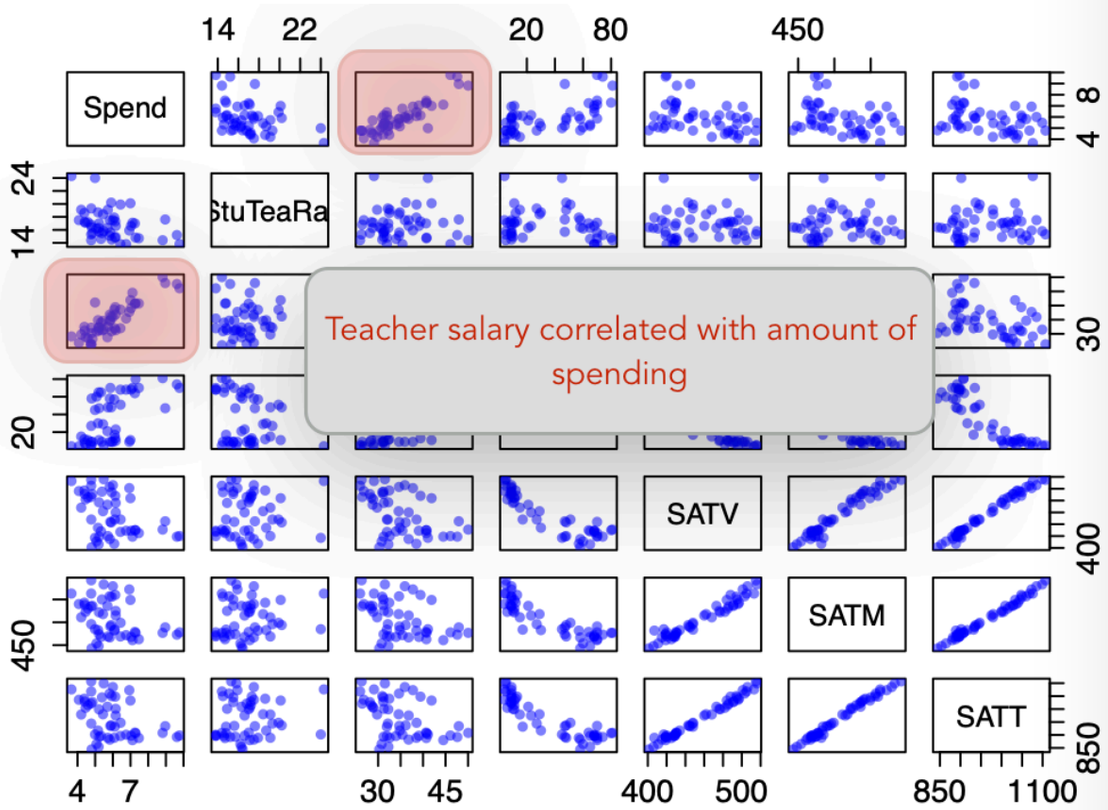
- All SAT scores highly correlated,
  → Just use total score as predicted variable



SAT Score Analysis

1: Initial plots

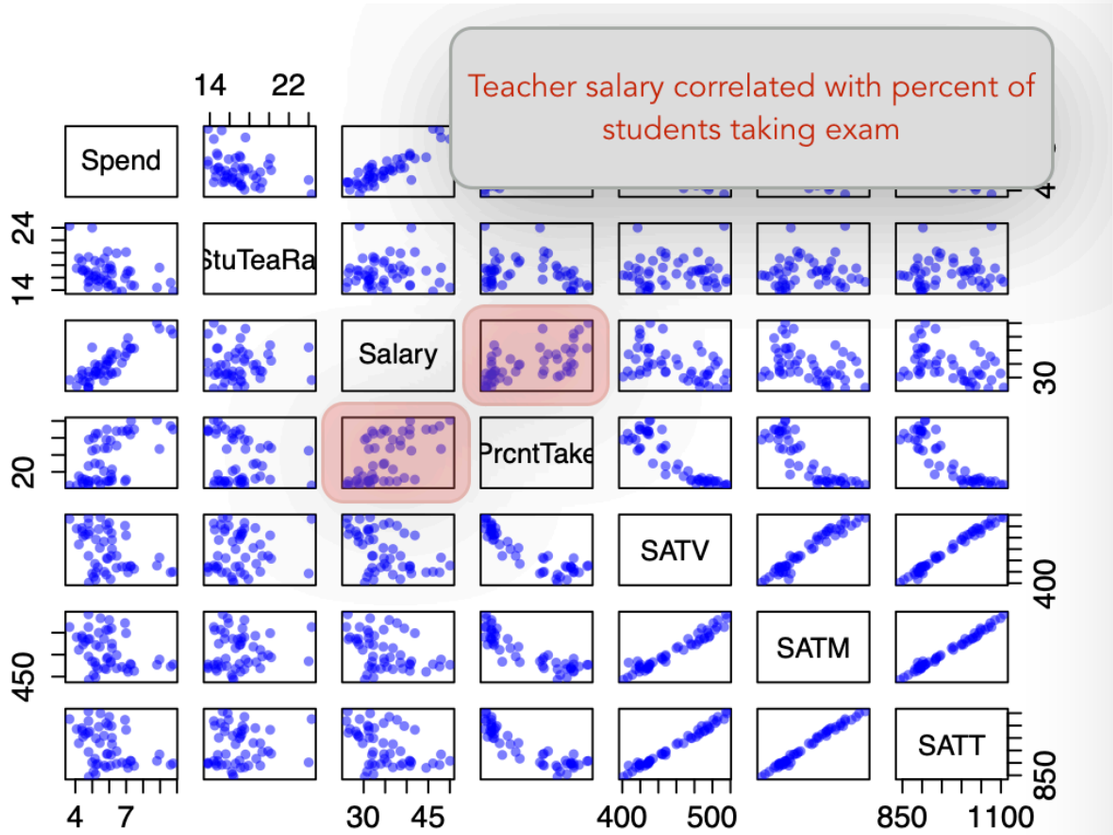
- All SAT scores highly correlated,
  → Just use total score as predicted variable
- Negative relationship between %Take and Scores



SAT Score Analysis

1: Initial plots

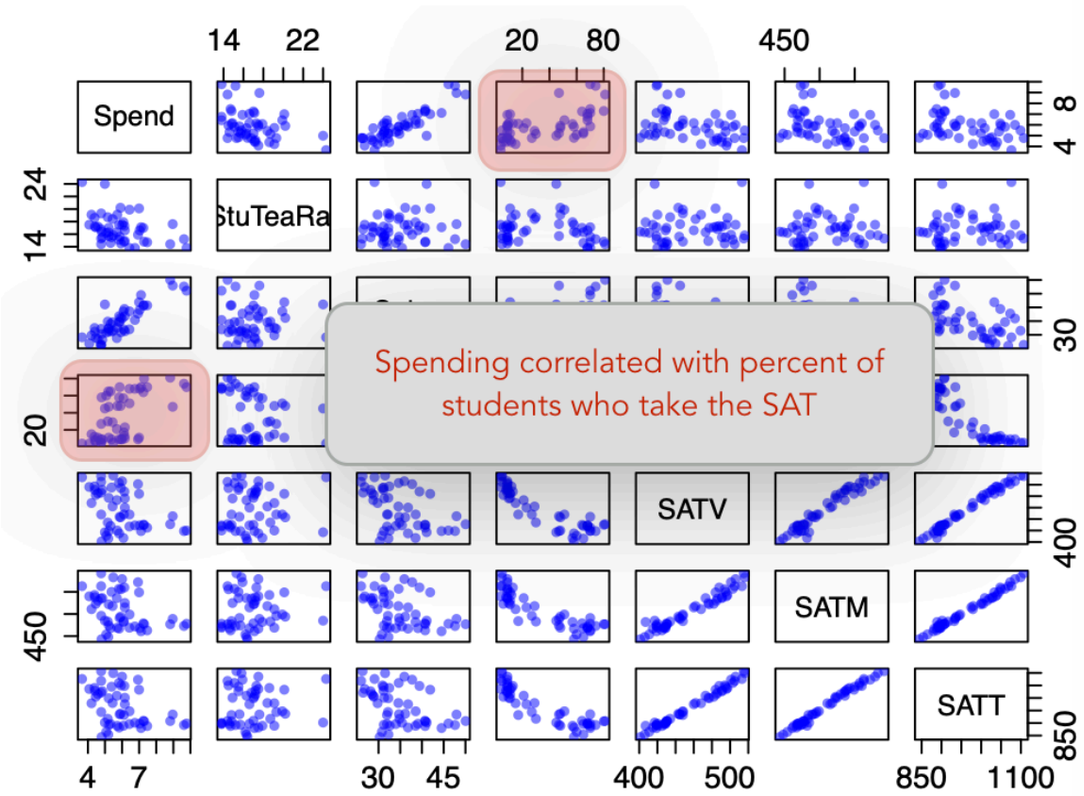
- All SAT scores highly correlated,
  → Just use total score as predicted variable
- Negative relationship between %Take and Scores
- Positive relationship between Teacher salary and total spending (Duh!)



SAT Score Analysis

1: Initial plots

- All SAT scores highly correlated,
  → Just use total score as predicted variable
- Negative relationship between %Take and Scores
- Positive relationship between Teacher salary and total spending (Duh!)
- Positive relationship between Teacher salary %Take



SAT Score Analysis

1: Initial plots

- All SAT scores highly correlated,
  → Just use total score as predicted variable
- Negative relationship between %Take and Scores
- Positive relationship between Teacher salary and total spending (Duh!)
- Positive relationship between Teacher salary %Take
- Positive relationship between Spending and %Take

SAT Score Analysis

1: Initial plots

- All SAT scores highly correlated,
  → Just use total score as predicted variable
- Negative relationship between %Take and Scores
- Positive relationship between Teacher salary and total spending (Duh!)
- Positive relationship between Teacher salary %Take
- Positive relationship between Spending and %Take
- Negative relationship Teacher Salary and SAT scores (??!!)

Data

- Are only interested in the following predictor variables:
  - Spend (column 2)
  - StuTeaRat (column 3)
  - Salary (column 4)
  - PrcntTake (column 5)

Data

```
cor(sat[, 2:5])
```

```
                Spend    StuTeaRat        Salary  PrcntTake
Spend       1.0000000 -0.371025386  0.869801513  0.5926274
StuTeaRat -0.3710254  1.000000000 -0.001146081 -0.2130536
Salary      0.8698015 -0.001146081  1.000000000  0.6167799
PrcntTake   0.5926274 -0.213053607  0.616779867  1.0000000
```

Data

```
cor(sat[, 2:5])
```

```
                Spend    StuTeaRat        Salary  PrcntTake
Spend       1.0000000 -0.371025386  0.869801513  0.5926274
StuTeaRat -0.3710254  1.000000000 -0.001146081 -0.2130536
Salary      0.8698015 -0.001146081  1.000000000  0.6167799
PrcntTake   0.5926274 -0.213053607  0.616779867  1.0000000
```

- `Spend` and `Salary` are fairly highly correlated
- Will try 3 models
  1. Including both
  2. Removing `spend` (just `salary`)
  3. Removing `salary` (just `spend`)

SAT Score Analysis

II: Correlation of Predictors

- Spending and teacher salary are highly correlated (~0.87)

→ Will run model 3 times:

a. once with all variables;
b. once with spending removed
c. once with salary removed

Full Model

Equation

$$y \sim \text{normal}(mu, sigma)$$

$$mu \sim \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \beta_3 x_3 + \beta_4 x_4$$

Equation

$$y \sim \text{normal}(mu, sigma)$$

$$mu \sim \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \beta_3 x_3 + \beta_4 x_4$$

Data: observed SAT scores

Data: spending amounts

Data: student:teacher ratios

Data: teacher salaries

Data: % of students take SAT

Equation

$$y \sim \text{normal}(mu, sigma)$$

Parameter: unexplained variation in the data after accounting for predictor variables

$$mu \sim \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \beta_3 x_3 + \beta_4 x_4$$

Parameter: average SAT score when all predictors are 0

Parameter: effect of spending on SAT scores

Parameter: effect of student:teacher ratios on SAT scores

Parameter: effect of teacher salaries on SAT scores

Parameter: effect of % of taking test on SAT scores

Load Libraries

```
library(rstan)
library(ggplot2)
library(loo)
```

Organize The Data

```
#--- SAT Scores ---#
y     = sat$SATT
yMean = mean(y)
ySD   = sd(y)
zy    = (y - yMean) / ySD

N = length(y)

#--- Spending ---#
x1     = sat$Spend
x1Mean = mean(x1)
x1SD   = sd(x1)
zx1    = (x1 - x1Mean) / x1SD
```

Organize The Data

```
#--- Student:Teacher Ratio ---#
x2     = sat$StuTeaRat
x2Mean = mean(x2)
x2SD   = sd(x2)
zx2    = (x2 - x2Mean) / x2SD

#--- Teacher Salary ---#
x3     = sat$Salary
x3Mean = mean(x3)
x3SD   = sd(x3)
zx3    = (x3 - x3Mean) / x3SD

#--- Precent Students Taking SAT ---#
x4     = sat$PrcntTake
x4Mean = mean(x4)
x4SD   = sd(x4)
zx4    = (x4 - x4Mean) / x4SD
```

Prepare the Data for Stan

```
dataList = list(
  y  = zy,
  x1 = zx1,
  x2 = zx2,
  x3 = zx3,
  x4 = zx4,
  N  = N
)
```

Build the Model
The data block

```
modelstring = "
  data {
     int N;            // Sample size
     vector[N] y;      // Vector of SAT scores
     vector[N] x1;     // Vector of Spending values
     vector[N] x2;     // Vector of Student:Teacher ratios
     vector[N] x3;     // Vector of Teacher Salaries
     vector[N] x4;     // Vector of percentage of students taking exams
  }
```

Build the Model

The `parameters` block

```
parameters {
    real b0;                    // Coefficient for 'intercept'
    real b1;                    // Coefficient for effect of spending
    real b2;                    // Coefficient for effect of student:teacher ratios
    real b3;                    // Coefficient for effect of teacher salaries
    real b4;                    // Coefficient for effect of % students taking SAT
    real<lower=0> sigma;        // Coefficient for sd for unexplained variation
}
```

Build the Model
The `model` block

```
model {
    // Definitions
    vector[N] mu;

    // Likelihood
    mu = b0 + b1*x1 + b2*x2 + b3*x3 + b4*x4;
    y ~ normal(mu, sigma);

    // Priors
    b0 ~ normal(0, 1);
    b1 ~ normal(0, 1);
    b2 ~ normal(0, 1);
    b3 ~ normal(0, 1);
    b4 ~ normal(0, 1);
    sigma ~ cauchy(1, 1);
}
```

Build the Model

The `generated quantities` block

```
generated quantities {
    // Posterior Predictive Variable Definitions
    vector[N] mu_pred;
    vector[N] y_pred;

    // WAIC Variable Definitions
    vector[N] log_lik;
    vector[N] mu_waic;

    // For Posterior Predictive Calculations
    for (i in 1:N) {
      mu_pred[i] = b0 + b1*x1[i] + b2*x2[i] + b3*x3[i] + b4*x4[i];
      y_pred[i] = normal_rng(mu_pred[i], sigma);
    }

    // For WAIC Calculatons
    for (i in 1:N) {
      mu_waic[i] = b0 + b1*x1[i] + b2*x2[i] + b3*x3[i] + b4*x4[i];
      log_lik[i] = normal_lpdf(y[i] | mu_waic[i], sigma);
    }
  }
"
writeLines(modelstring, con = “model1.stan")
```

Build the Model

The `generated quantities` block

```
generated quantities {
   // Posterior Predictive Variable Definitions
   vector[N] mu_pred;
   vector[N] y_pred;

   // WAIC Variable Definitions
   vector[N] log_lik;
   vector[N] mu_waic;

   // For Posterior Predictive Calculations
   for (i in 1:N) {
     mu_pred[i] = b0 + b1*x1[i] + b2*x2[i] + b3*x3[i] + b4*x4[i];
     y_pred[i] = normal_rng(mu_pred[i], sigma);
   }

   // For WAIC Calculatons
   for (i in 1:N) {
     mu_waic[i] = b0 + b1*x1[i] + b2*x2[i] + b3*x3[i] + b4*x4[i];
     log_lik[i] = normal_lpdf(y[i] | mu_waic[i], sigma);
   }
  }
"
writeLines(modelstring, con = “model1.stan")
```

For posterior predictive check

Build the Model
The `generated quantities` block

```
generated quantities {
    // Posterior Predictive Variable Definitions
    vector[N] mu_pred;
    vector[N] y_pred;

    // WAIC Variable Definitions
    vector[N] log_lik;
    vector[N] mu_waic;

    // For Posterior Predictive Calculations
    for (i in 1:N) {
      mu_pred[i] = b0 + b1*x1[i] + b2*x2[i] + b3*x3[i] + b4*x4[i];
      y_pred[i] = normal_rng(mu_pred[i], sigma);
    }

    // For WAIC Calculatons
    for (i in 1:N) {
      mu_waic[i] = b0 + b1*x1[i] + b2*x2[i] + b3*x3[i] + b4*x4[i];
      log_lik[i] = normal_lpdf(y[i] | mu_waic[i], sigma);
    }
  }
"
writeLines(modelstring, con = “model1.stan")
```

Required for
WAIC calculations

Build the Model

The `generated quantities` block

```
generated quantities {
    // Posterior Predictive Variable Definitions
    vector[N] mu_pred;
    vector[N] y_pred;

    // WAIC Variable Definitions
    vector[N] log_lik;
    vector[N] mu_waic;

    // For Posterior Predictive Calculations
    for (i in 1:N) {
      mu_pred[i] = b0 + b1*x1[i] + b2*x2[i] + b3*x3[i] + b4*x4[i];
      y_pred[i] = normal_rng(mu_pred[i], sigma);
    }

    // For WAIC Calculatons
    for (i in 1:N) {
      mu_waic[i] = b0 + b1*x1[i] + b2*x2[i] + b3*x3[i] + b4*x4[i];
      log_lik[i] = normal_lpdf(y[i] | mu_waic[i], sigma);
    }
  }
"
writeLines(modelstring, con = “model1.stan")
```

Variable **must** have this name to work!!!

Build the Model

The `generated quantities` block

```
generated quantities {
    // Posterior Predictive Variable Definitions
    vector[N] mu_pred;
    vector[N] y_pred;

    // WAIC Variable Definitions
    vector[N] log_lik;
    vector[N] mu_waic;

    // For Posterior Predictive Calcu
    for (i in 1:N) {
      mu_pred[i] = b0 + b1*x1[i] + b2                    i];
      y_pred[i] = normal_rng(mu_pred[i], sigma);
    }

    // For WAIC Calculatons
    for (i in 1:N) {
      mu_waic[i] = b0 + b1*x1[i] + b2*x2[i] + b3*x3[i] + b4*x4[i];
      log_lik[i] = normal_lpdf(y[i] | mu_waic[i], sigma);
    }
  }
"
writeLines(modelstring, con = “model1.stan")
```

Calculates the log probability density function

Build the Model

The generated quantities block

```
generated quantities {
    // Posterior Predictive Variable Definitions
    vector[N] mu_pred;
    vector[N] y_pred;

    // WAIC Variable Definitions
    vector[N] log_lik;
    vector[N] mu_waic;

    // For Posterior Predictive Calcu
    for (i in 1:N) {
      mu_pred[i] = b0 + b1*x1[i] + b2                    i];
      y_pred[i] = normal_rng(mu_pred[i], sigma);
    }

    // For WAIC Calculatons
    for (i in 1:N) {
      mu_waic[i] = b0 + b1*x1[i] + b2*x2[i]  b3*x3[i] + b4*x4[i];
      log_lik[i] = normal_lpdf(y[i] | mu_waic[i], sigma);
    }
  }
"
writeLines(modelstring, con = “model1.stan")
```

Requires the equation in this format!

Build the Model

The `generated quantities` block

```
generated quantities {
    // Posterior Predictive Variable Definitions
    vector[N] mu_pred;
    vector[N] y_pred;

    // WAIC Variable Definitions
    vector[N] log_lik;
    vector[N] mu_waic;

    // For Posterior Predictive Calculation
    for (i in 1:N) {
      mu_pred[i] = b0 + b1*x1[i] + b2*x2[i]
      y_pred[i] = normal_rng(mu_pred[i], sig
    }

    // For WAIC Calculatons
    for (i in 1:N) {
      mu_waic[i] = b0 + b1*x1[i] + b2*x2[i] + b3*x3[i] + b4*x4[i];
      log_lik[i] = normal_lpdf(y[i] | mu_waic[i], sigma);
    }
  }
"
writeLines(modelstring, con = "model1.stan")
```

Since we will be running multiple models, calling this one "`model1.stan`"

Run the Model

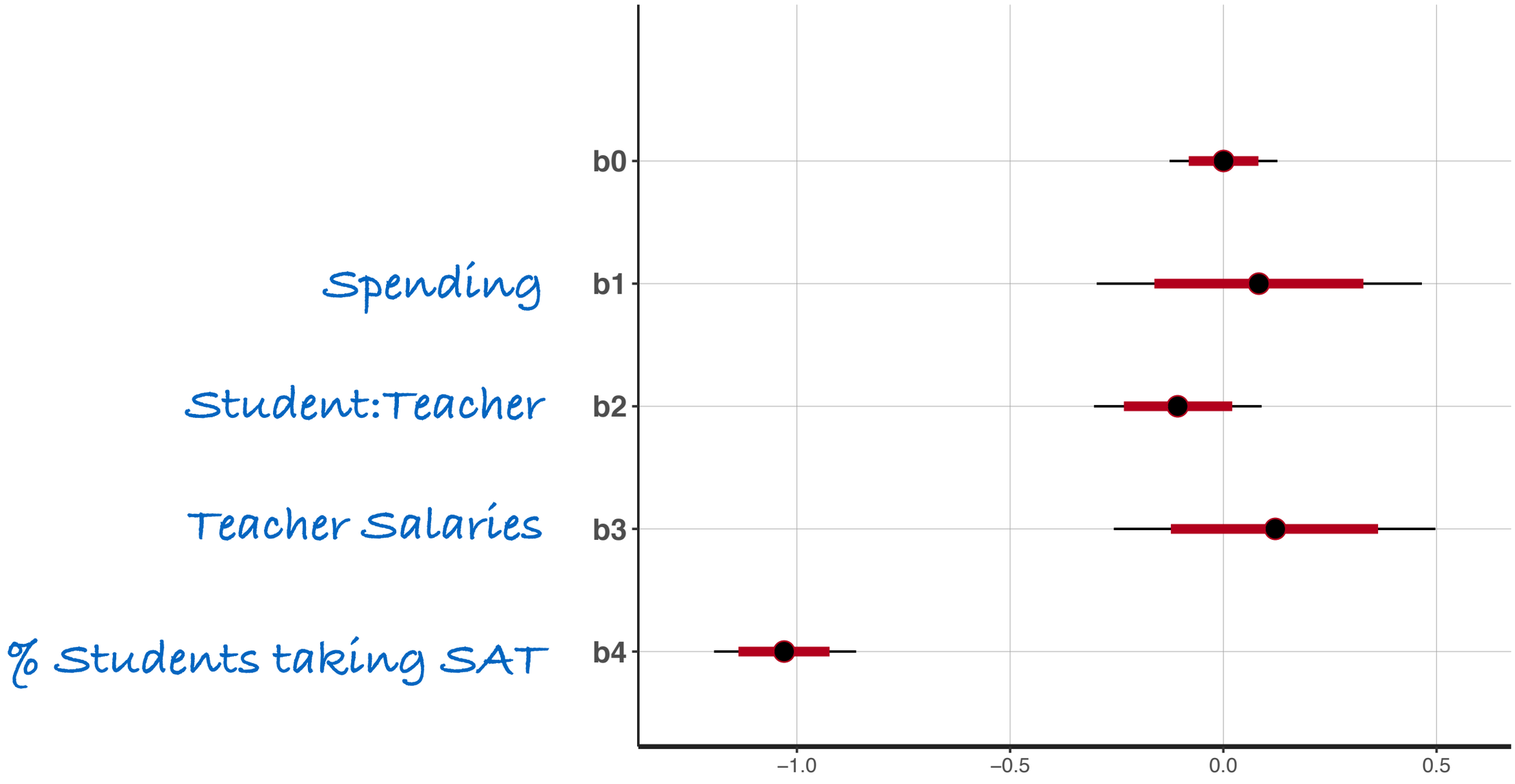
```
model1 = stan(file = "model1.stan",
              data = dataList,
              pars = c("b0", "b1", "b2", "b3", "b4", "sigma", "y_pred", "log_lik"),
              warmup = 2000,
              iter = 7000,
              chains = 3)
```

Check MCMC Performance

```
print(model1)

Inference for Stan model: model1.
3 chains, each with iter=7000; warmup=2000; thin=1;
post-warmup draws per chain=5000, total post-warmup draws=15000.

             mean se_mean   sd  2.5%   25%   50%   75% 97.5% n_eff Rhat
b0           0.00    0.00 0.06 -0.13 -0.04  0.00  0.04  0.13 13252    1
b1           0.08    0.00 0.19 -0.30 -0.04  0.08  0.21  0.47  5731    1
b2          -0.11    0.00 0.10 -0.30 -0.17 -0.11 -0.04  0.09  6296    1
b3           0.12    0.00 0.19 -0.26 -0.01  0.12  0.25  0.50  5575    1
b4          -1.03    0.00 0.08 -1.19 -1.09 -1.03 -0.97 -0.86 10952    1
sigma        0.45    0.00 0.05  0.37  0.42  0.45  0.48  0.56 11284    1
y_pred[1]    0.86    0.00 0.48 -0.07  0.55  0.86  1.18  1.78 13675    1
y_pred[2]   -0.03    0.00 0.49 -1.00 -0.35 -0.03  0.30  0.95 15253    1
...
```

Plot Results

```
stan_plot(model1, par = c("b0", "b1", "b2", "b3", "b4"))
```



Calculate WAIC

```
loglik1 = extract_log_lik(model1)
waic1 = waic(loglik1)
waic1

Computed from 15000 by 50 log-likelihood matrix

          Estimate   SE
elpd_waic    -33.5  5.7
p_waic         5.7  1.6
waic          66.9 11.5
```

Calculate WAIC

Expected log pointwise predictive density

```
loglik1 = extract_log_lik(model1)
waic1 = waic(loglik1)
waic1

Computed from 15000 by 50 log-likelihood matrix

          Estimate   SE
elpd_waic    -33.5  5.7
p_waic         5.7  1.6
waic          66.9 11.5
```

Calculate WAIC

```
loglik1 = extract_log_lik(model1)
waic1 = waic(loglik1)
waic1

Computed from 15000 by 50 log-likelihood matrix

          Estimate   SE
elpd_waic    -33.5  5.7
p_waic         5.7  1.6
waic          66.9 11.5
```

Effective number of parameters (note that we had 6 parameters)

Calculate WAIC

```
loglik1 = extract_log_lik(model1)
waic1 = waic(loglik1)
waic1

Computed from 15000 by 50 log-likelihood matrix

          Estimate   SE
elpd_waic    -33.5  5.7
p_waic         5.7  1.6
waic          66.9 11.5
```

The information criterion

Model With
Spending Removed

Build the Model

`x1` and `b1` removed

```
modelstring = "
  data {
    int N;           // Sample size
    vector[N] y;     // Vector of SAT scores
    vector[N] x2;    // Vector of Student:Teacher ratios
    vector[N] x3;    // Vector of Teacher Salaries
    vector[N] x4;    // Vector of percentage of students taking exams
  }
  parameters {
    real b0;                  // Coefficient for 'intercept'
    real b2;                  // Coefficient for effect of student:teacher ratios
    real b3;                  // Coefficient for effect of teacher salaries
    real b4;                  // Coefficient for effect of % students taking SAT
    real<lower=0> sigma;      // Coefficient for sd for unexplained variation
  }
```

Build the Model

```
model {
  // Definitions
  vector[N] mu;

  // Likelihood
  mu = b0 + b2*x2 + b3*x3 + b4*x4;
  y ~ normal(mu, sigma);

  // Priors
  b0 ~ normal(0, 1);
  b2 ~ normal(0, 1);
  b3 ~ normal(0, 1);
  b4 ~ normal(0, 1);
  sigma ~ cauchy(1, 1);
}
```

`x1` and `b1` removed

Build the Model

`x1` and `b1` removed

```
generated quantities {
  // Posterior Predictive Variable Definitions
  vector[N] mu_pred;
  vector[N] y_pred;

  // WAIC Variable Definitions
  vector[N] log_lik;
  vector[N] mu_waic;

  // For Posterior Predictive Calculations
  for (i in 1:N) {
    mu_pred[i] = b0 + b2*x2[i] + b3*x3[i] + b4*x4[i];
    y_pred[i] = normal_rng(mu_pred[i], sigma);
  }

  // For WAIC Calculatons
  for (i in 1:N) {
    mu_waic[i] = b0 + b2*x2[i] + b3*x3[i] + b4*x4[i];
    log_lik[i] = normal_lpdf(y[i] | mu_waic[i], sigma);
  }
 }
"
writeLines(modelstring, con = "model2.stan")
```

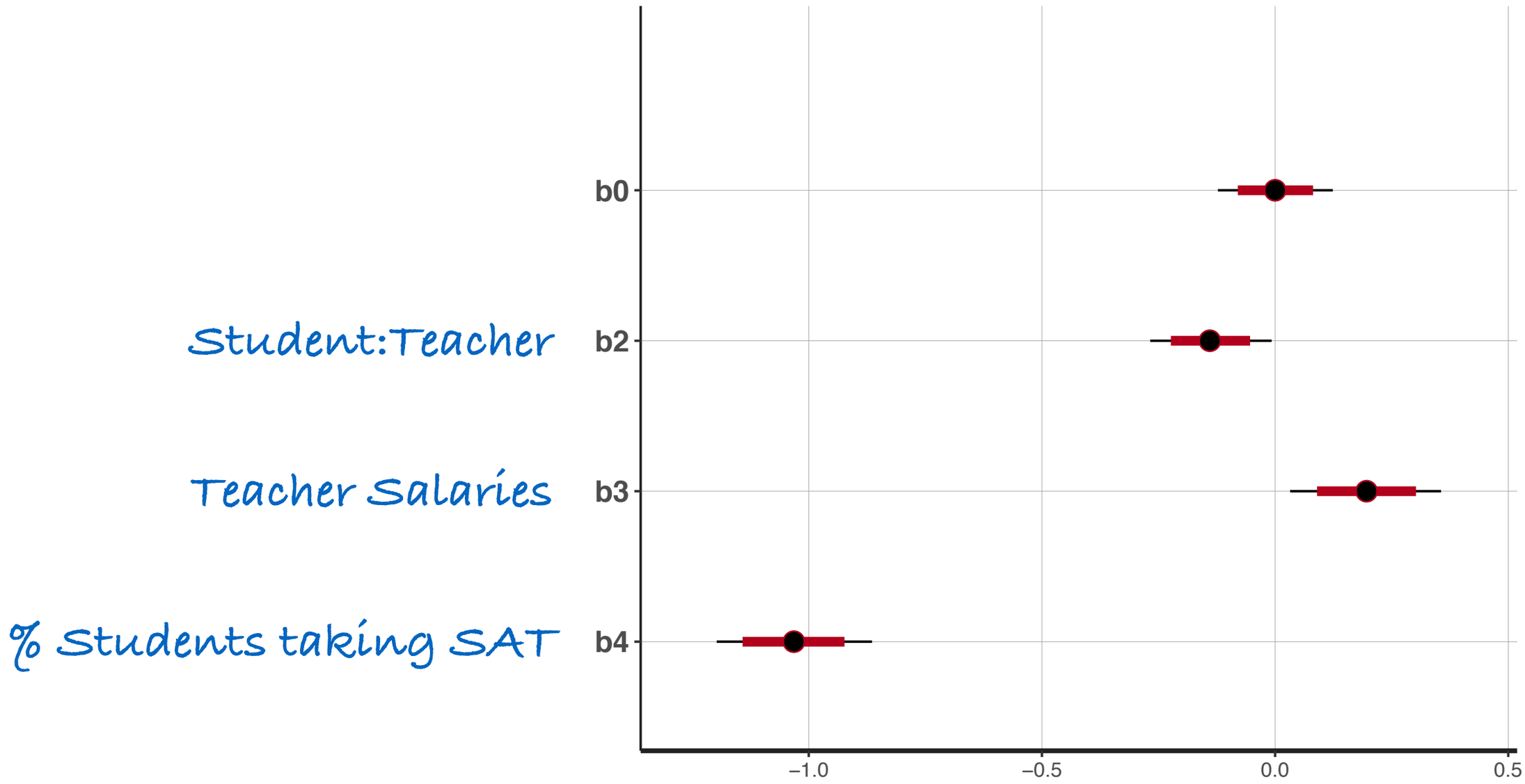
Run the Model

b1 removed

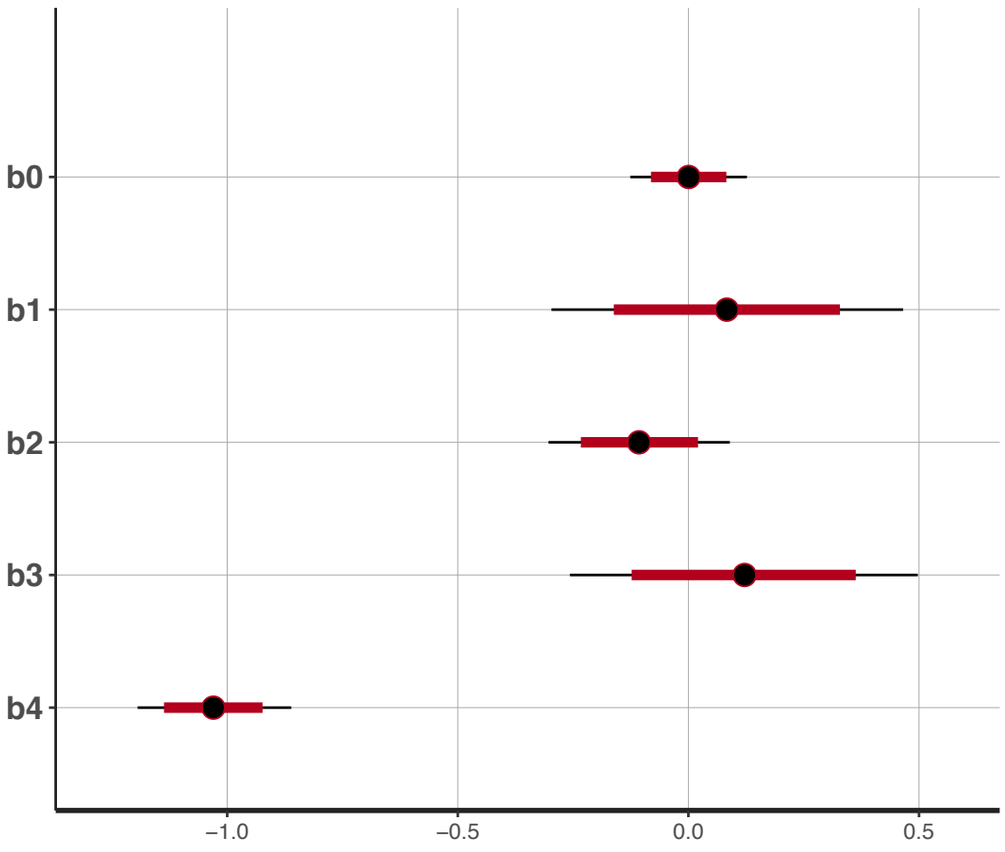
```
model2 = stan(file = "model2.stan",
              data = dataList,
              pars = c("b0", "b2", "b3", "b4", "sigma", "y_pred", "log_lik"),
              warmup = 2000,
              iter = 7000,
              chains = 3)
```

Plot Results

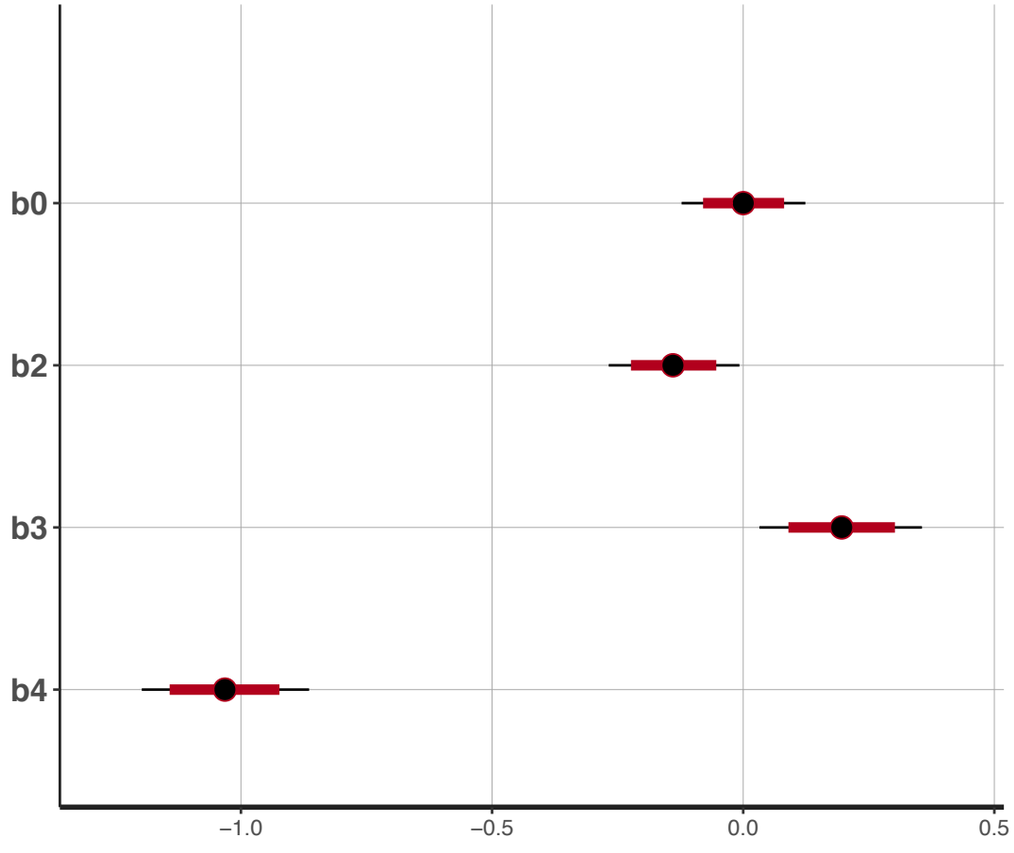
```
stan_plot(model2, par = c("b0", "b2", "b3", "b4"))
```



Full Model



Spending Removed



Calculate WAIC

```
loglik2 = extract_log_lik(model2)
waic2 = waic(loglik2)
```

Model With `Salary` Removed

Build the Model

x3 and b3 removed

```
modelstring = "
  data {
    int N;           // Sample size
    vector[N] y;     // Vector of SAT scores
    vector[N] x1;    // Vector of Spending values
    vector[N] x2;    // Vector of Student:Teacher ratios
    vector[N] x4;    // Vector of percentage of students taking exams
  }
  parameters {
    real b0;                  // Coefficient for 'intercept'
    real b1;                  // Coefficient for effect of spending
    real b2;                  // Coefficient for effect of student:teacher ratios
    real b4;                  // Coefficient for effect of % students taking SAT
    real<lower=0> sigma;      // Coefficient for unexplained variance
  }
```

Build the Model

`x3` and `b3` removed

```
model {
  // Definitions
  vector[N] mu;

  // Likelihood
  mu = b0 + b1*x1 + b2*x2 + b4*x4;
  y ~ normal(mu, sigma);

  // Priors
  b0 ~ normal(0, 1);
  b1 ~ normal(0, 1);
  b2 ~ normal(0, 1);
  b4 ~ normal(0, 1);
  sigma ~ cauchy(1, 1);
}
```

Build the Model

x3 and b3 removed

```
generated quantities {
  // Posterior Predictive Variable Definitions
  vector[N] mu_pred;
  vector[N] y_pred;

  // WAIC Variable Definitions
  vector[N] log_lik;
  vector[N] mu_waic;

  // For Posterior Predictive Calculations
  for (i in 1:N) {
    mu_pred[i] = b0 + b1*x1[i] + b2*x2[i] + b4*x4[i];
    y_pred[i] = normal_rng(mu_pred[i], sigma);
  }

  // For WAIC Calculatons
  for (i in 1:N) {
    mu_waic[i] = b0 + b1*x1[i] + b2*x2[i] + b4*x4[i];
    log_lik[i] = normal_lpdf(y[i] | mu_waic[i], sigma);
  }
 }
"
writeLines(modelstring, con = "model3.stan")
```

Run the Model

b3 removed

```
model3 = stan(file = "model3.stan",
              data = dataList,
              pars = c("b0", "b1", "b2", "b4", "sigma", "y_pred", "log_lik"),
              warmup = 2000,
              iter = 7000,
              chains = 3)
```

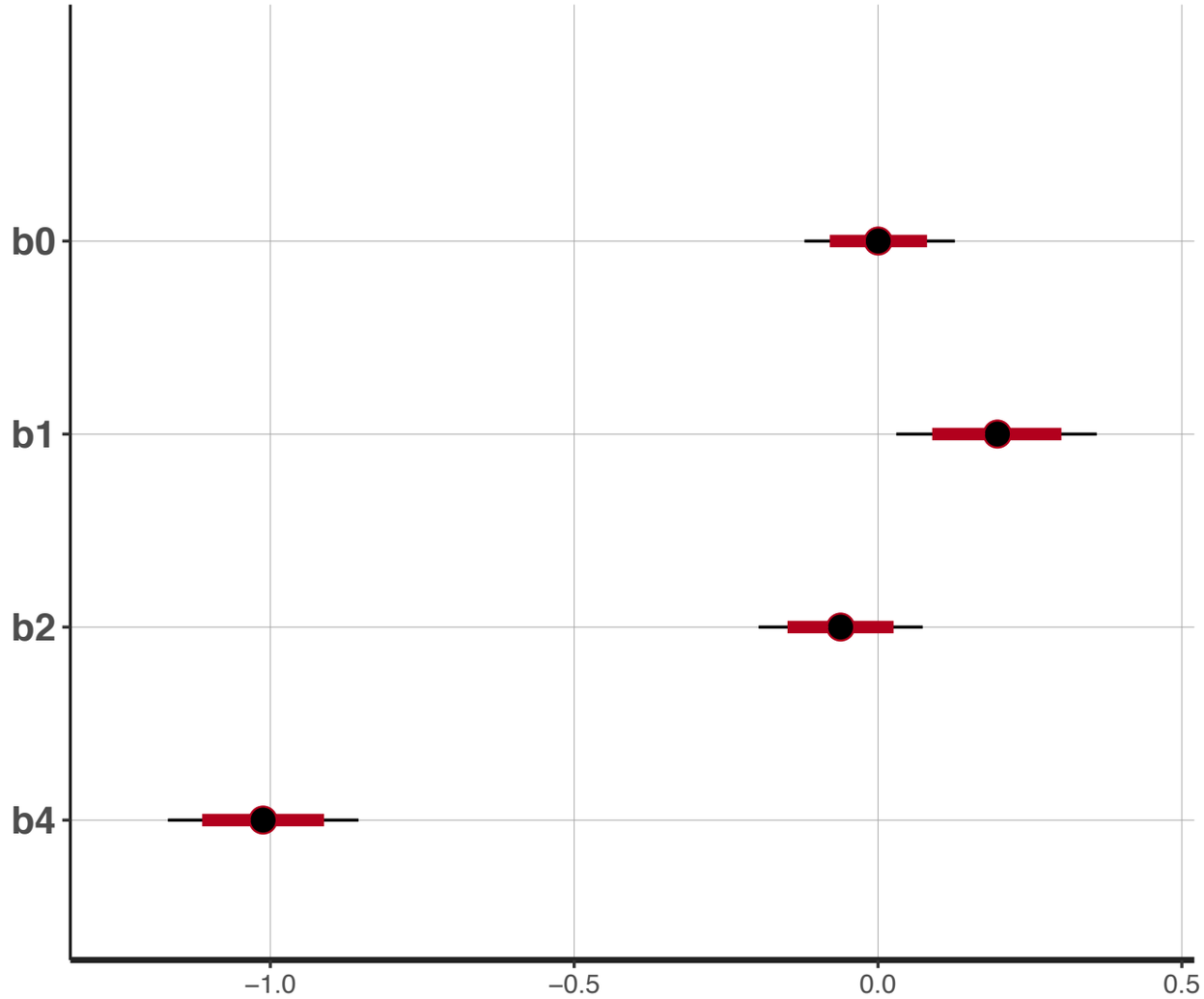
Plot Results

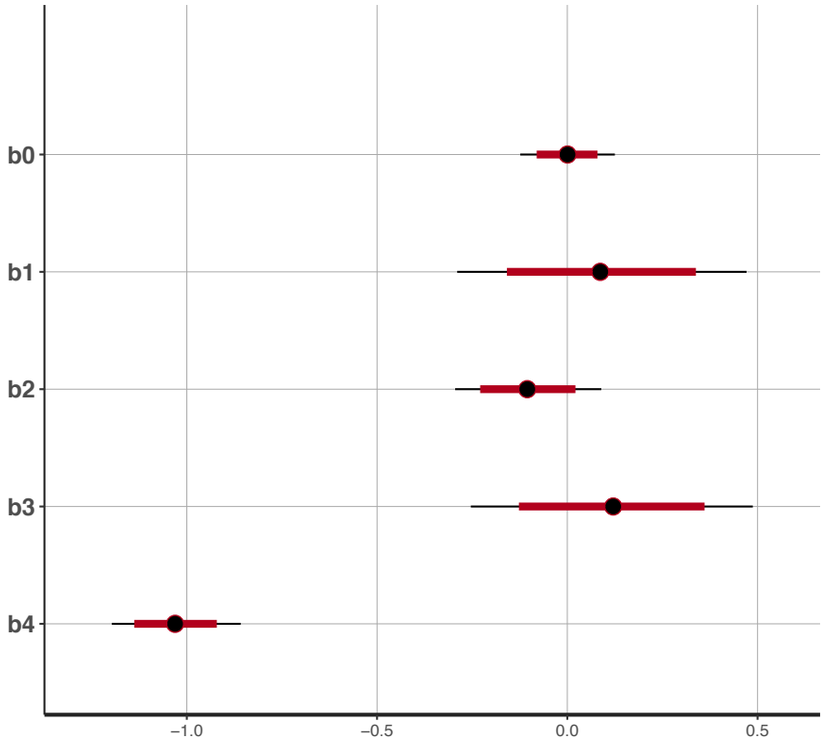
```
stan_plot(model3, par = c("b0", "b1", "b2", "b4"))
```

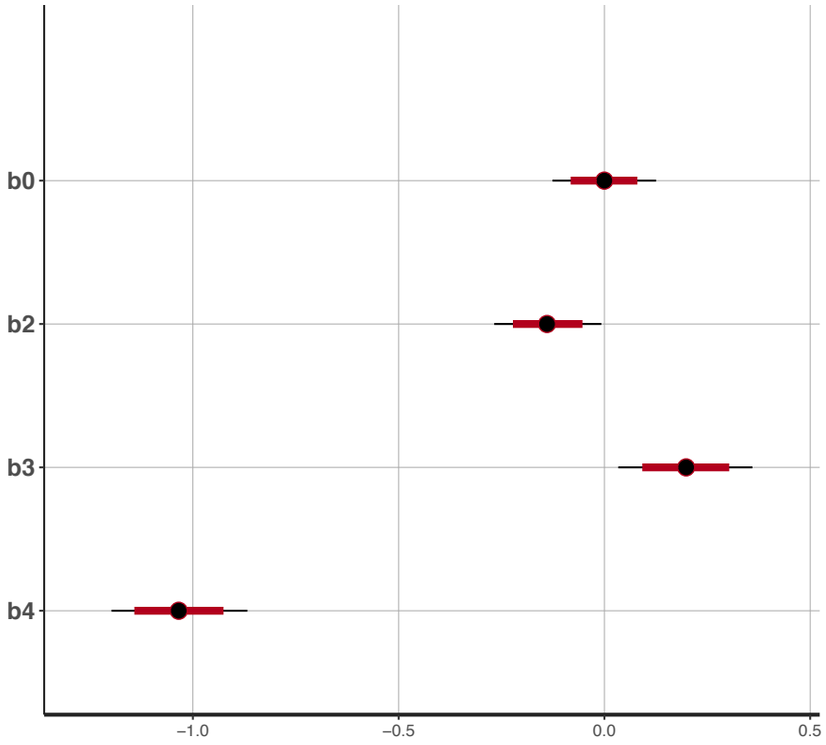
Spending

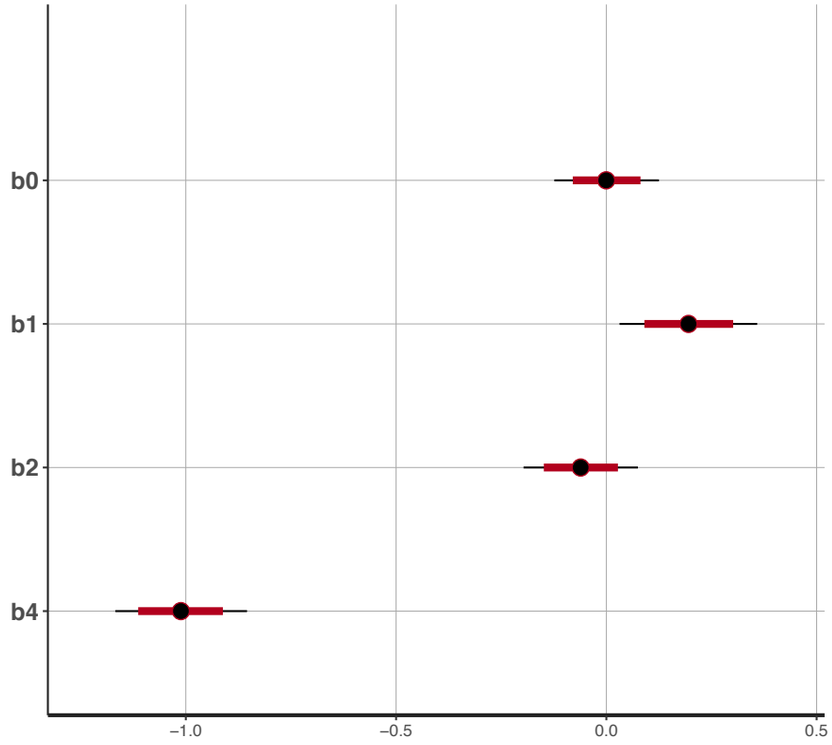
Student:Teacher

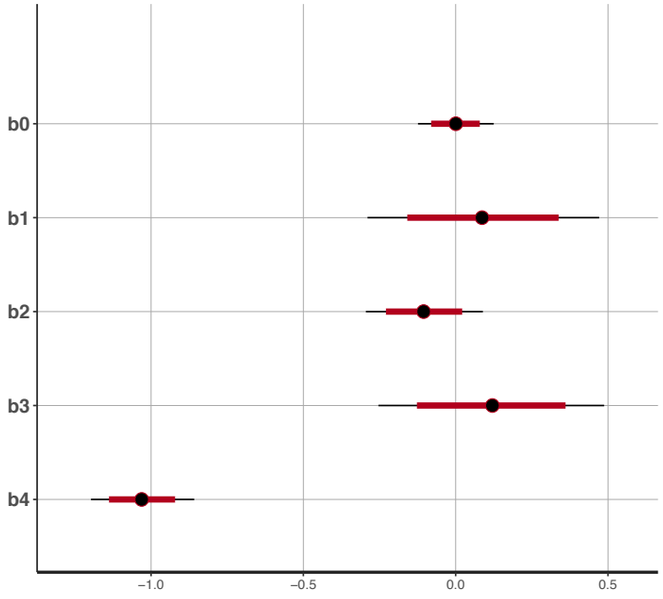
% Students taking SAT

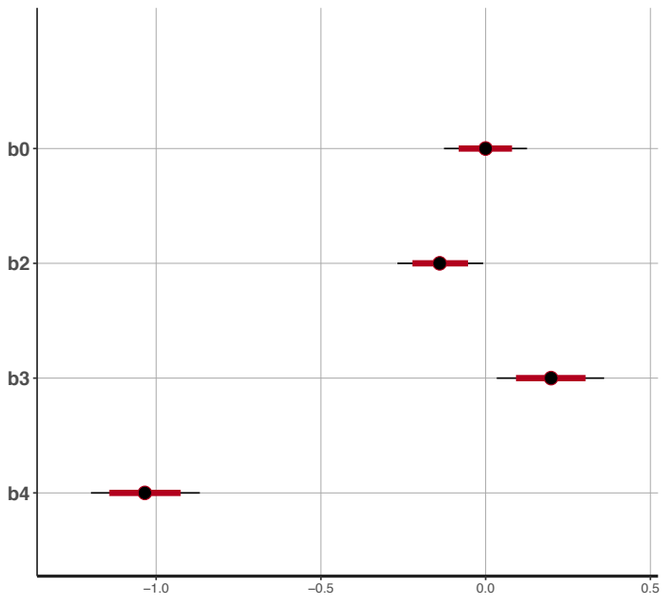


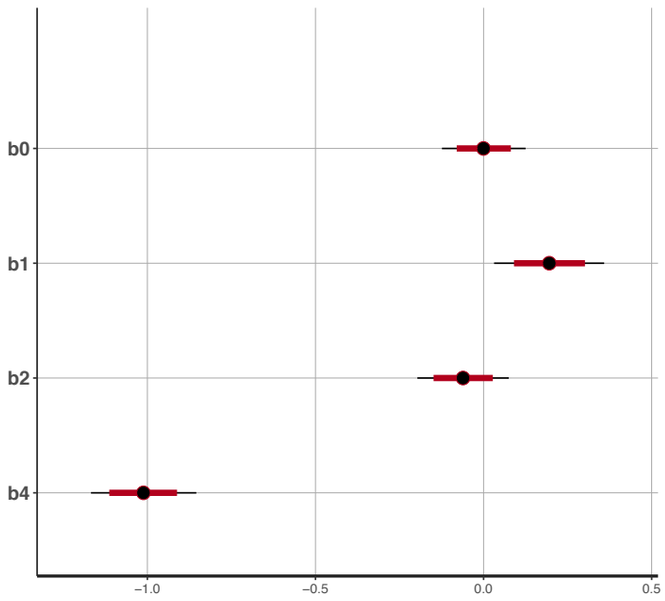

b0
b1
b2
b3
b4
−1.0
−0.5
0.0
0.5


b0
b2
b3
b4
−1.0
−0.5
0.0
0.5


b0
b1
b2
b4
−1.0
−0.5
0.0
0.5







SAT Score Analysis

III: Model Results I: Posterior Plots

- % students taking SAT clearly has the strongest effect, quite negative
  - estimated effect insensitive to removing parameters
- Removing Spending reduces variation around effects of both student:teacher and salary
  - Is correlated with both, masking some effects
- Removing Salary reduces variation around effects of Spending

  Removing Spending results in clearest picture of other effects
- Effect of Salary is now positive!!!

Calculate WAIC

```
loglik3 = extract_log_lik(model3)
waic3 = waic(loglik3)
```

Compare WAIC Scores

```
compare(waic1, waic2, waic3)

      elpd_diff elpd_waic se_elpd_waic p_waic se_p_waic   waic  se_waic
waic3   0.0      -32.7        5.7           5.0    1.4      65.4  11.4
waic2   0.0      -32.7        5.8           5.2    1.6      65.4  11.6
waic1  -0.8      -33.5        5.7           5.7    1.6      66.9  11.5
```

Compare WAIC Scores

```
compare(waic1, waic2, waic3)

      elpd_diff elpd_waic se_elpd_waic p_waic se_p_waic   waic  se_waic
waic3    0.0      -32.7        5.7          5.0    1.4      65.4  11.4
waic2    0.0      -32.7        5.8          5.2    1.6      65.4  11.6
waic1   -0.8      -33.5        5.7          5.7    1.6      66.9  11.5
```

Lowest is “best”, but these are roughly equal, particularly given SE

Compare WAIC Scores

```
compare(waic1, waic2, waic3)

      elpd_diff elpd_waic se_elpd_waic p_waic se_p_waic   waic  se_waic
waic3    0.0      -32.7        5.7         5.0    1.4      65.4  11.4
waic2    0.0      -32.7        5.8         5.2    1.6      65.4  11.6
waic1   -0.8      -33.5        5.7         5.7    1.6      66.9  11.5
```

Model 3 has equal predictive power, but fewer parameters (though not really)

SAT Score Analysis

IV: Model Results II: WAIC

- All models have roughly equal WAIC scores
  - Confirms original correlation of Spending and Salary: leaving one out does not reduce model perfomance

Does **not** mean that these parameters do not have an effect!!!

SAT Score Analysis

IV: Model Results II: WAIC

- All models have roughly equal WAIC scores
  - Confirms original correlation of Spending and Salary: leaving one out does not reduce model perfomance

Interpreting Effects

Interpreting Effects

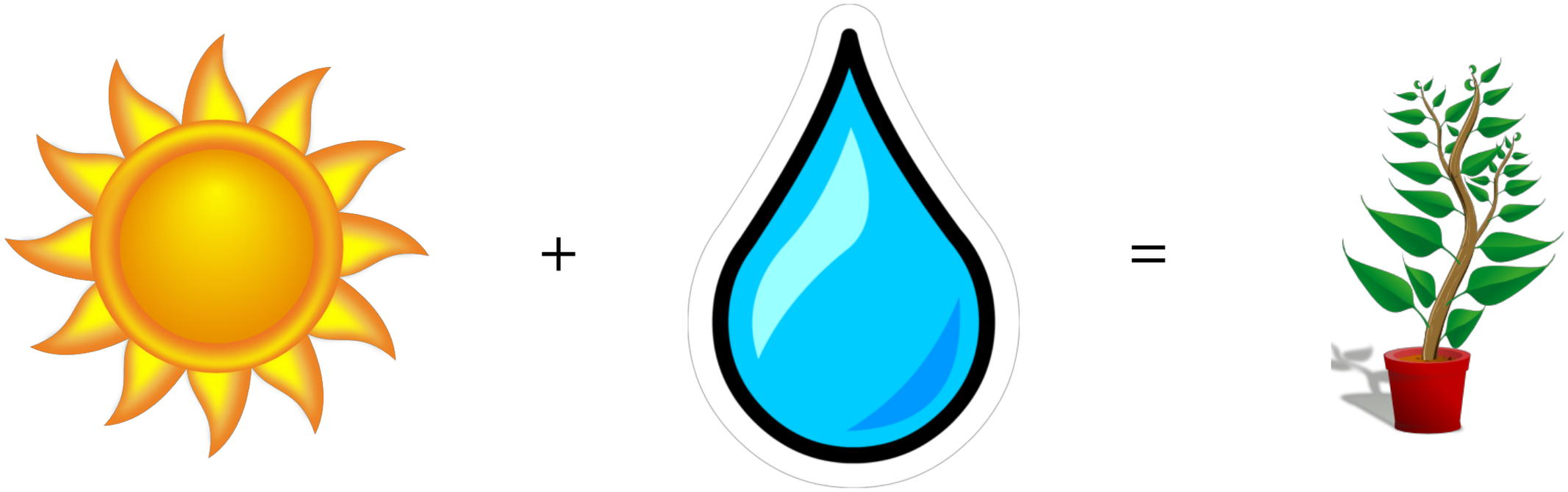
- Which predictor variables do you think are important?

Interpreting Effects

- Which predictor variables do you think are important?
- What is their effect on SAT scores?
  - Parameter estimates can be misleading
    - Effect depends on what values other parameters are

Interpreting Effects

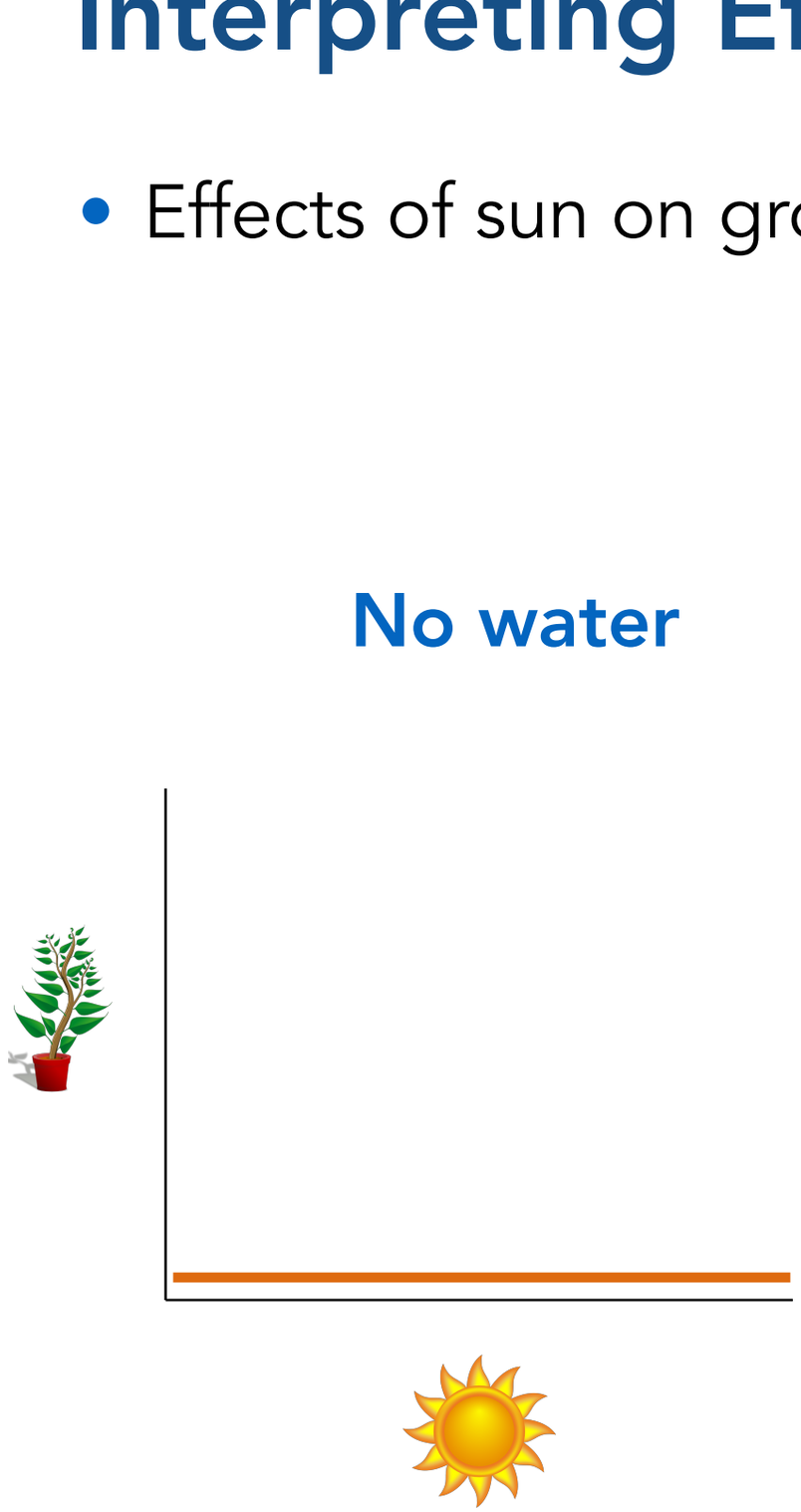
- Effect of water and sunlight on plant growth
  - Model will give you one value (with distribution) for each...
    but is this realistic?



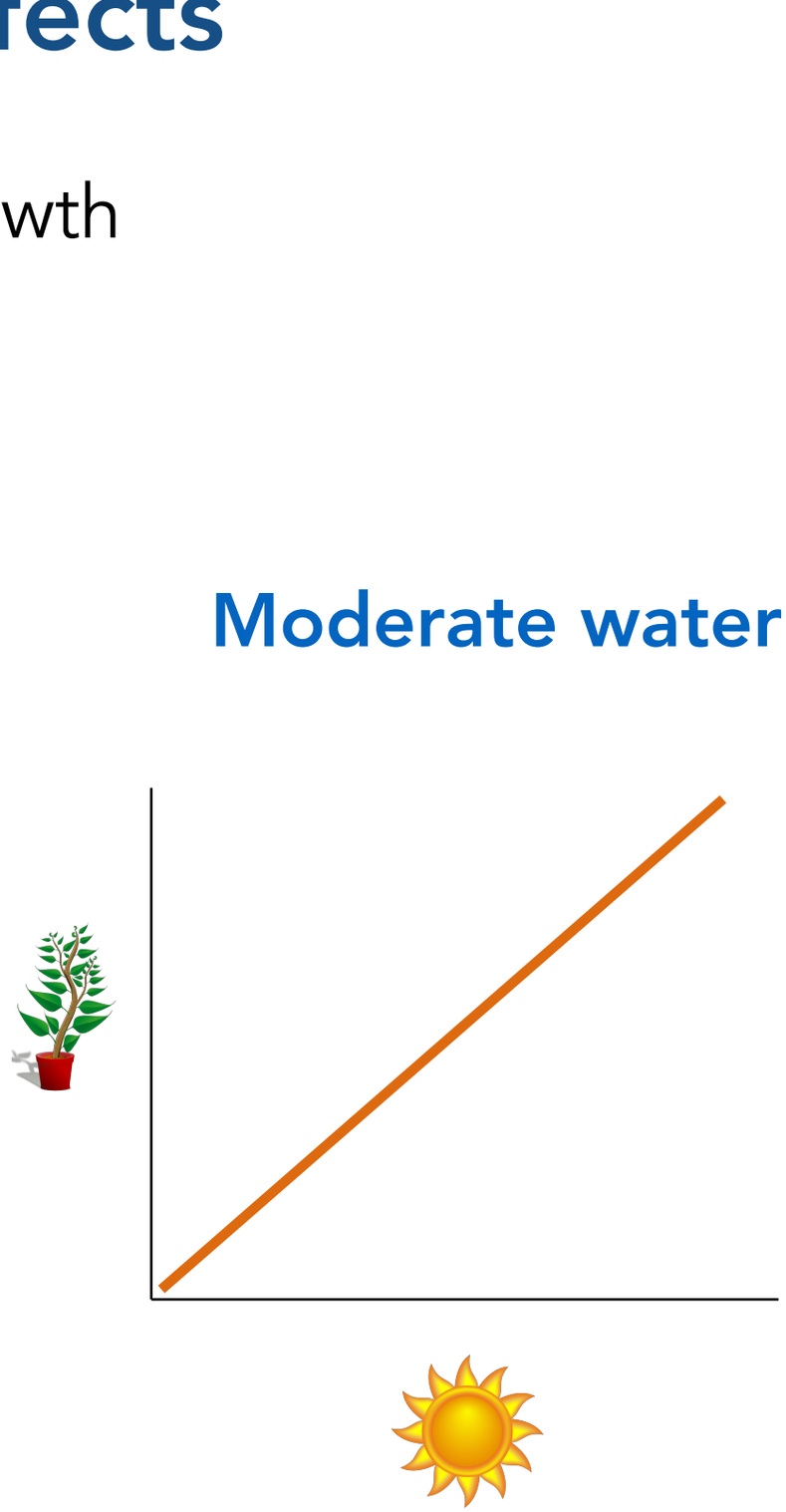
Interpreting Effects

- Effects of sun on growth

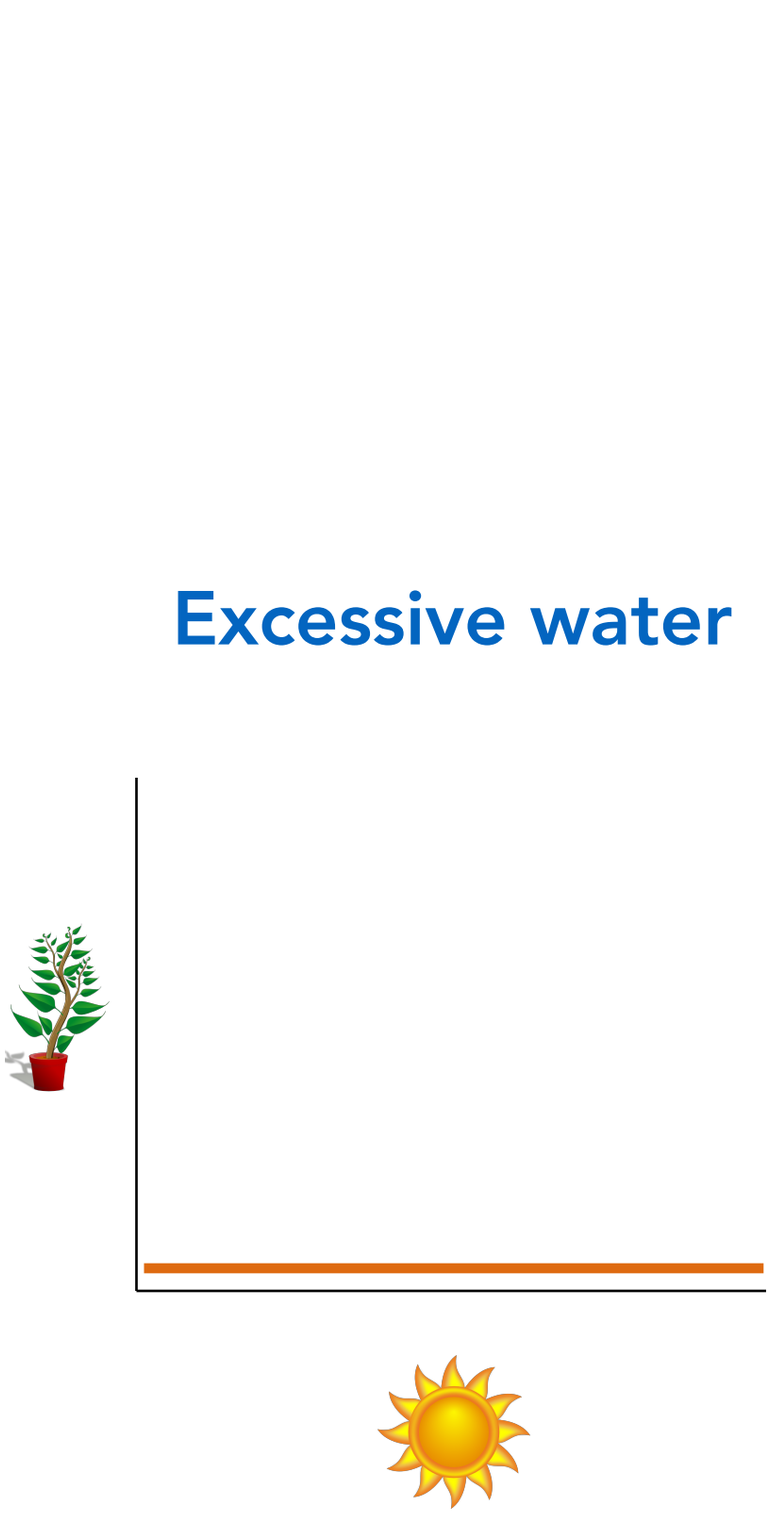
No water



Moderate water



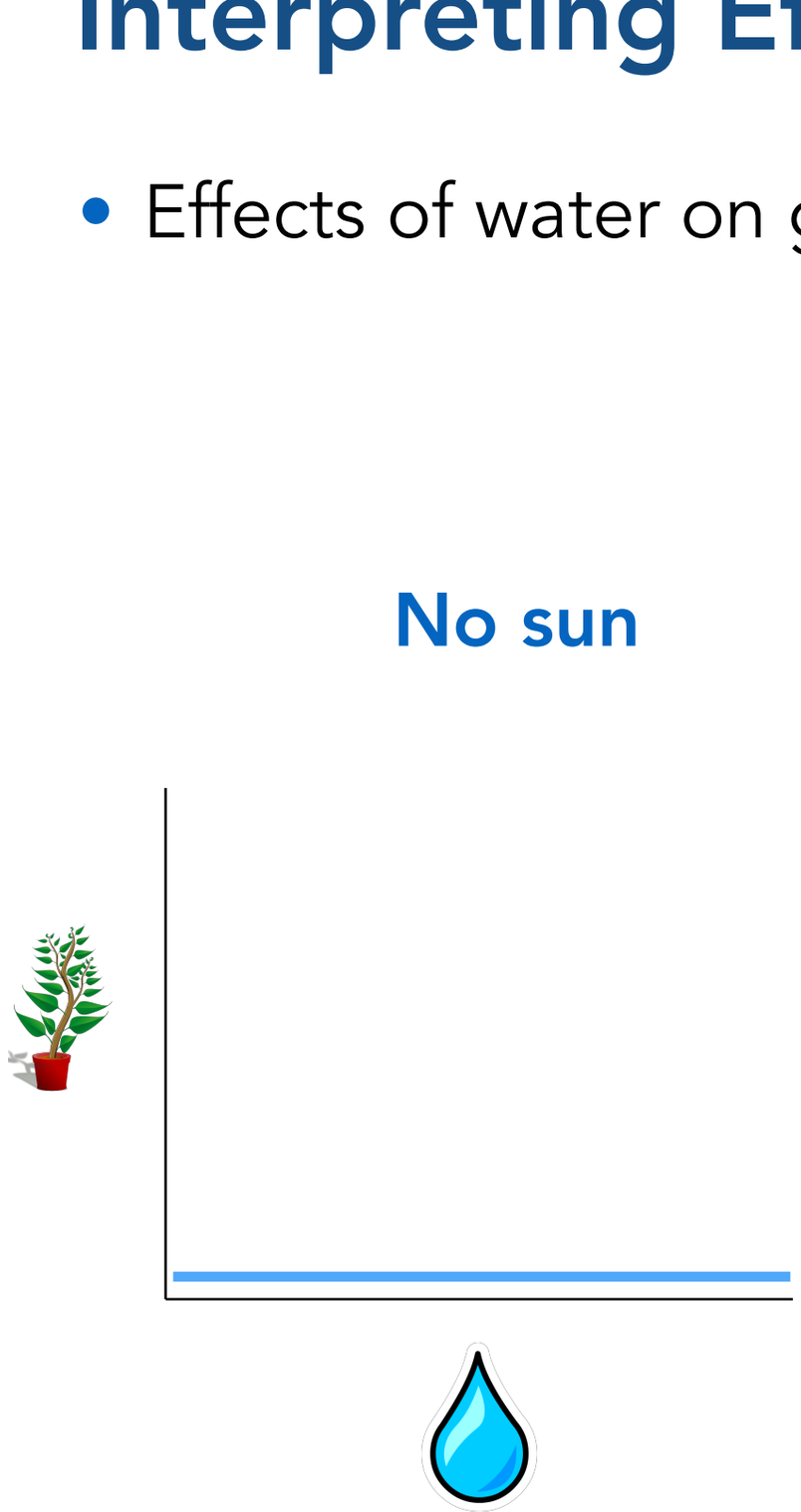
Excessive water



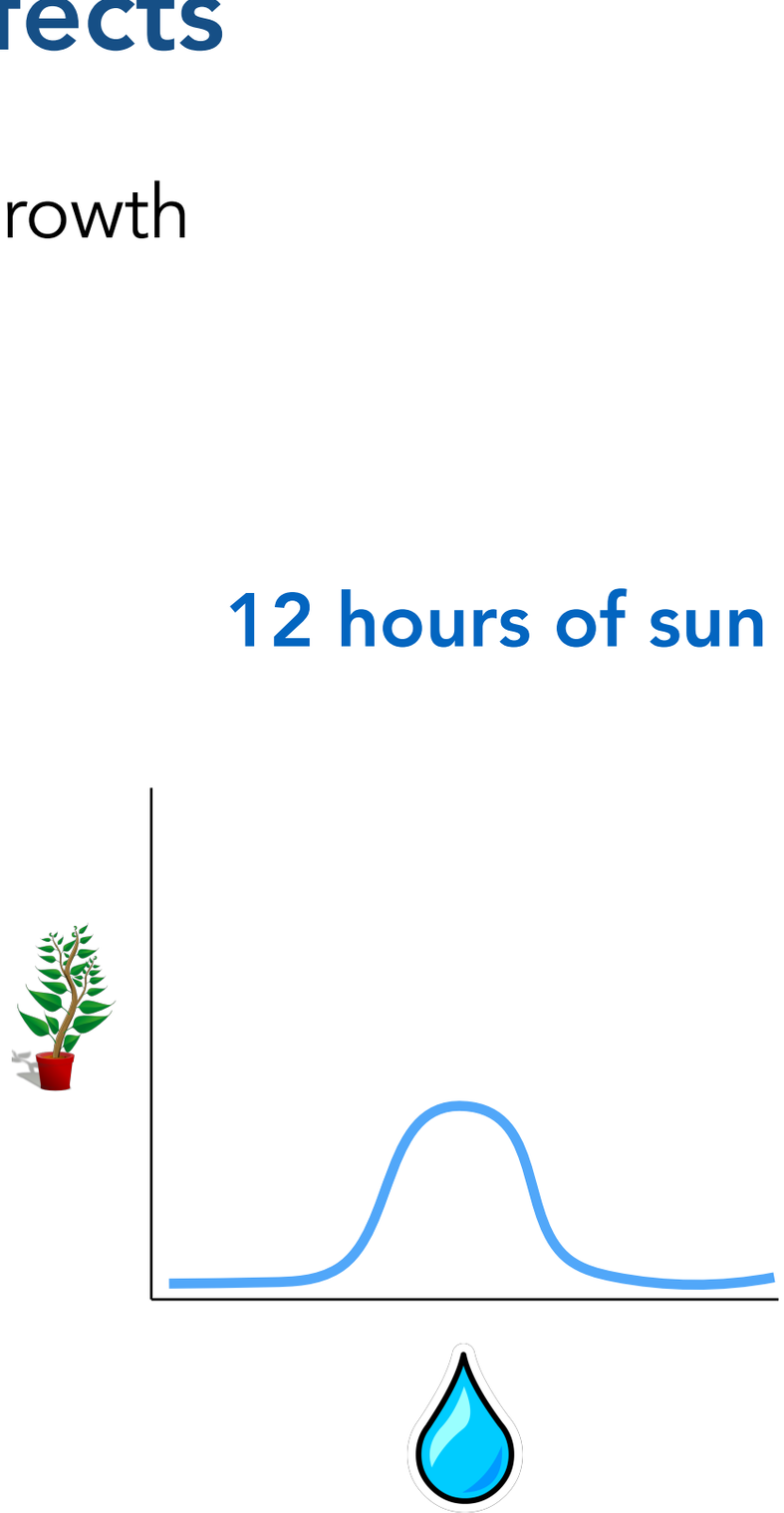
Interpreting Effects

- Effects of water on growth

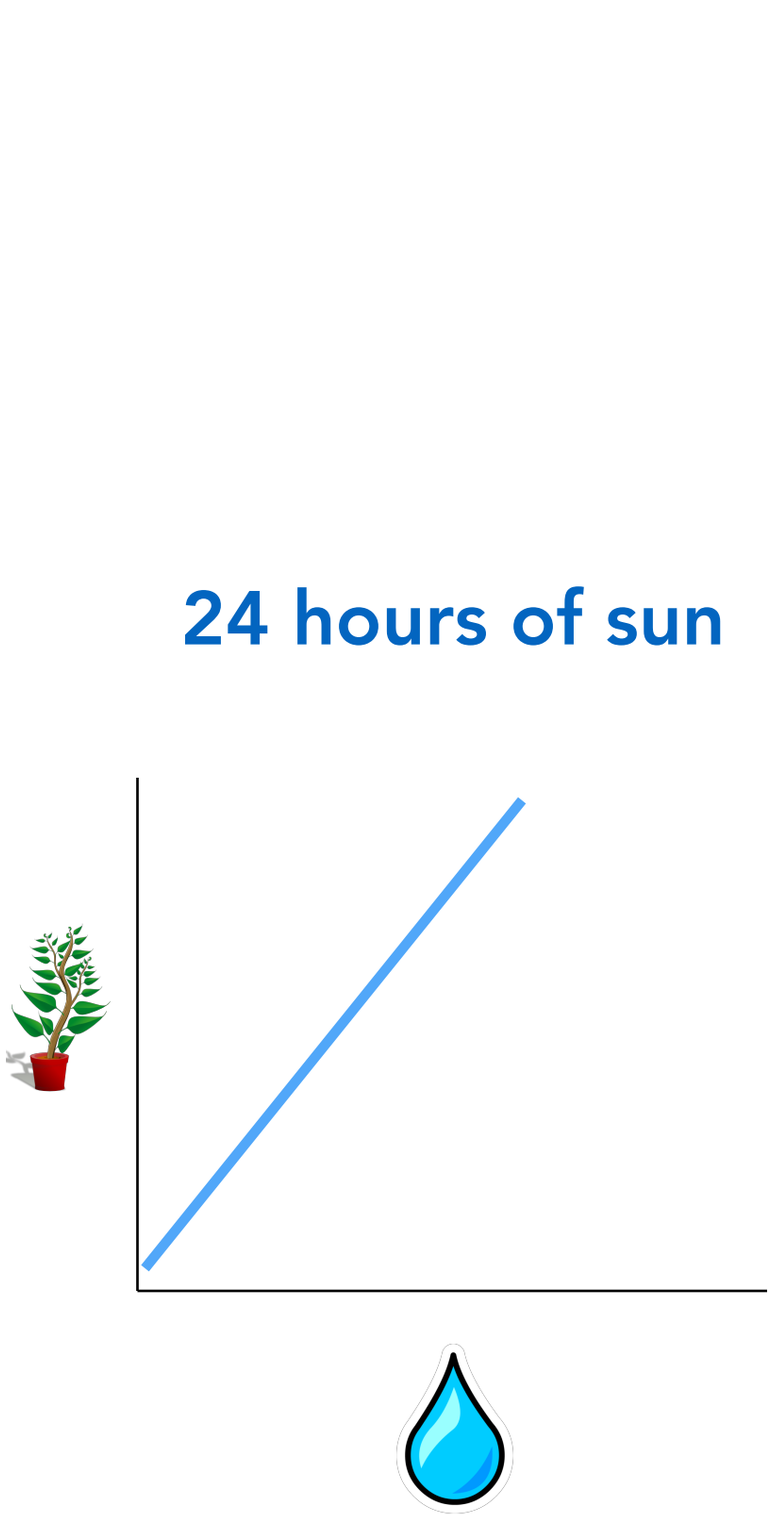
No sun



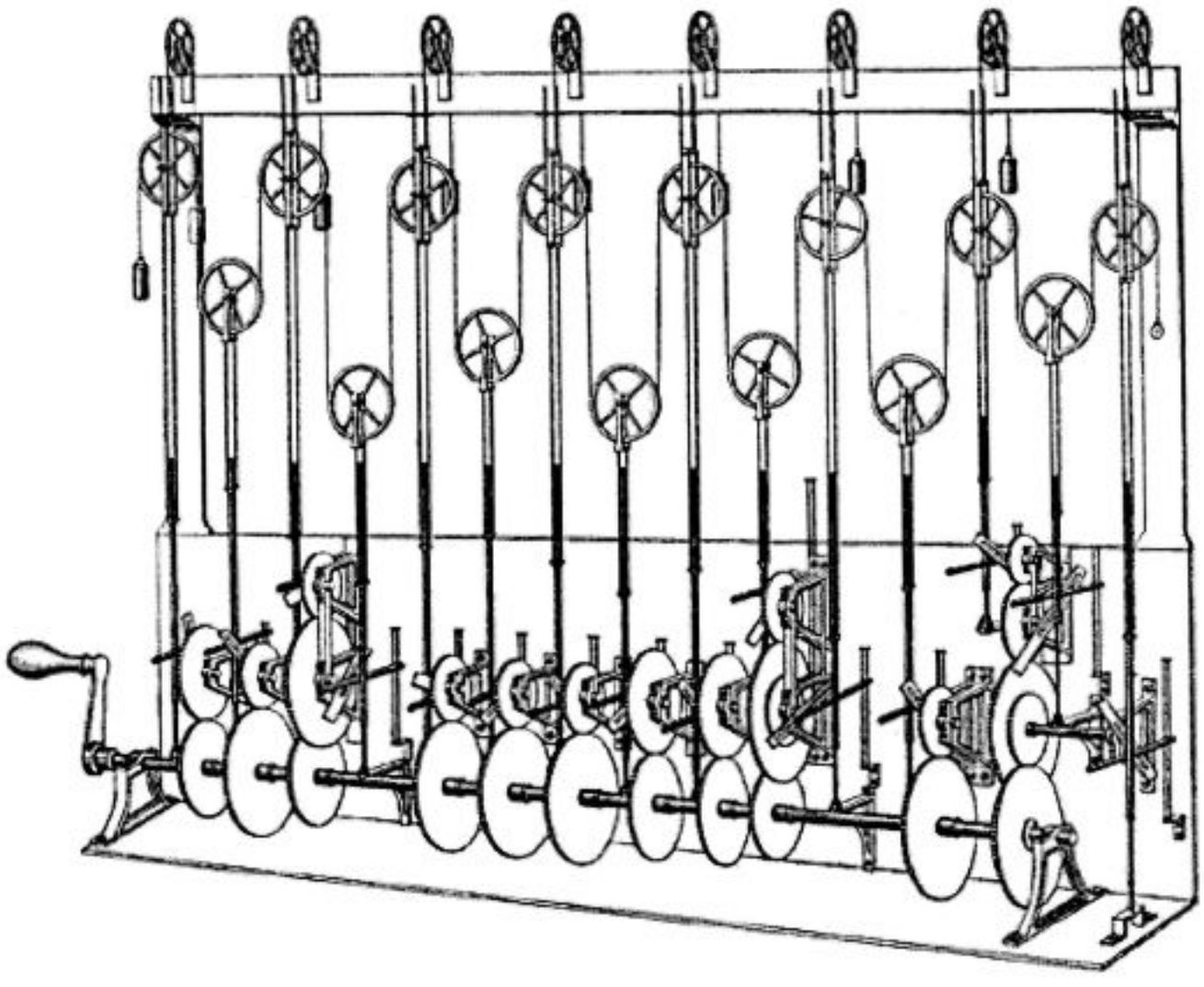
12 hours of sun



24 hours of sun



Interpreting Effects



Interpreting Effects

- For each parameter of interest, plot vs the predicted variable, with other parameters *at least* at their minimum, mean, and maximum values

- Can use model that is best representative of that parameter

Spending
Based on model3 (salary: zx3 and b3 removed)

- Retrieve posteriors from appropriate model

```
mcmcData = as.data.frame(model3)
zb0 = mcmcData$b0
zb1 = mcmcData$b1
zb2 = mcmcData$b2
zb4 = mcmcData$b4
```

Spending: minimums
Based on model3 (salary: zx3 and b3 removed)

- Calculate predicted `y` values, given the **minimum** values for all other parameters

```
spendingMins = matrix(0, nrow = length(zb0), ncol = length(zy))

for (i in 1:length(zb0)) {
  for (j in 1:length(zy)) {
    spendingMins[i, j] = zb0[i] + (zb1[i] * zx1[j]) + (zb2[i] * min(zx2)) + (zb4[i] * min(zx4))
  }
}
```

Spending: minimums

Based on model3 (sa oved)

- Calculate predicted `y` valu for all other parameters

`zx1` will vary across all observed values

```
spendingMins = matrix(0, nrow = length(zb0), ncol = length(zy))

for (i in 1:length(zb0)) {
  for (j in 1:length(zy)) {
    spendingMins[i, j] = zb0[i] + (zb1[i] * zx1[j]) + (zb2[i] * min(zx2)) + (zb4[i] * min(zx4))
  }
}
```

Spending: minimums

Based on model3 (salary: zx3 and b3 removed)

- Calculate predi**num** values for all other parameters

coefficient for other variables multiplied by the **minimum** value for each

```
spendingMins = matrix(0, nrow = length(zb0), ncol = length(zy))

for (i in 1:length(zb0)) {
  for (j in 1:length(zy)) {
    spendingMins[i, j] = zb0[i] + (zb1[i] * zx1[j]) + (zb2[i] * min(zx2)) + (zb4[i] * min(zx4))
  }
}
```

Spending: minimums
Based on model3 (salary: zx3 and b3 removed)

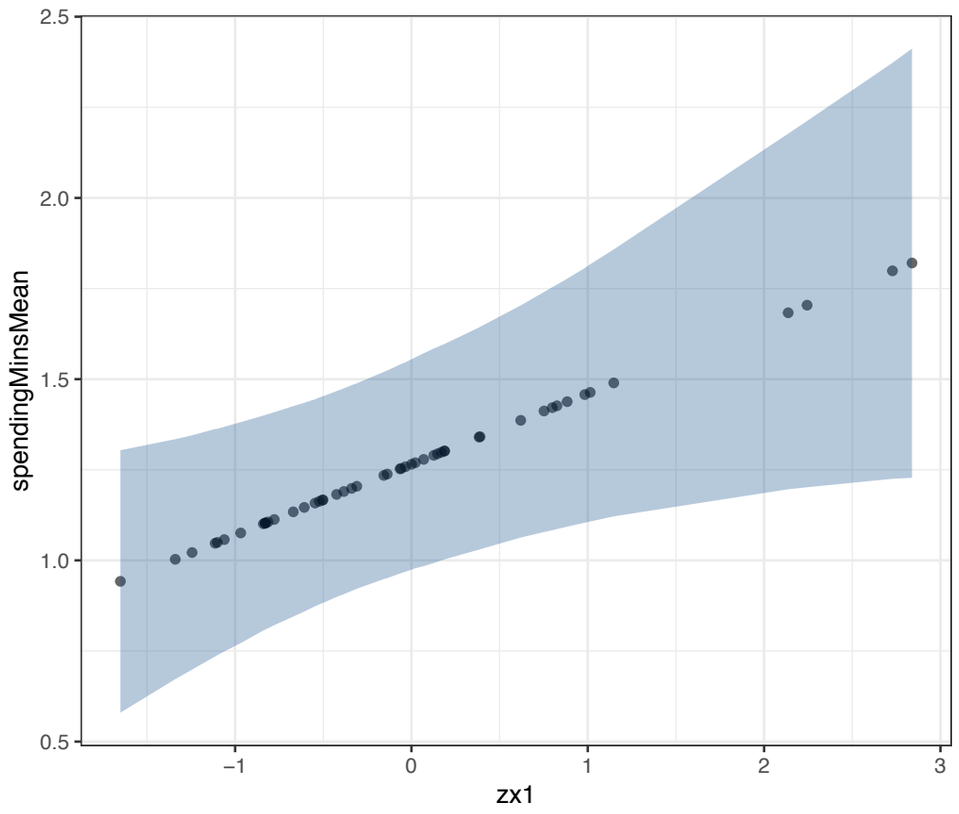
- Calculate mean predicted values, and associated HDI

```
spendingMeansMean = apply(spendingMeans, 2, mean)
spendingMeansLow = apply(spendingMeans, 2, quantile, probs = 0.025)
spendingMeansHigh = apply(spendingMeans, 2, quantile, probs = 0.975)
```

Spending: minimums
Based on model3 (salary: zx3 and b3 removed)

- Plot predictions

```
spendingMinsSummary = data.frame(zx1, spendingMinsMean, spendingMinsLow, spendingMinsHigh)

ggplot(spendingMinsSummary) +
  theme_bw() +
  geom_point(aes(x = zx1, y = spendingMinsMean), alpha = 0.6) +
  geom_ribbon(aes(x = zx1, ymin = spendingMinsLow, ymax = spendingMinsHigh), fill =
    “dodgerblue4", alpha = 0.3)
```



Spending: means
Based on model3 (salary: zx3 and b3 removed)

- Calculate predicted `y` values, given the **mean** values for all other parameters

```
spendingMeans = matrix(0, nrow = length(zb0), ncol = length(zy))

for (i in 1:length(zb0)) {
  for (j in 1:length(zy)) {
    spendingMeans[i, j] = zb0[i] + (zb1[i] * zx1[j]) + (zb2[i] * mean(zx2)) + (zb4[i] * mean(zx4))
  }
}
```

Spending: means

Based on model3 (salary: zx3 and b3 removed)

- Calculate predicted y val … all other parameters

`zx1` will vary across all observed values

```
spendingMeans = matrix(0, nrow = length(zb0), ncol = length(zy))

for (i in 1:length(zb0)) {
  for (j in 1:length(zy)) {
    spendingMeans[i, j] = zb0[i] + (zb1[i] * zx1[j]) + (zb2[i] * mean(zx2)) + (zb4[i] * mean(zx4))
  }
}
```

Spending: means

Based on model3 (salary: zx3 and b3 removed)

- Calculate predi values for all other parameters

coefficient for other variables multiplied by the **mean** value for each

```
spendingMeans = matrix(0, nrow = length(zb0), ncol = length(zy))

for (i in 1:length(zb0)) {
  for (j in 1:length(zy)) {
    spendingMeans[i, j] = zb0[i] + (zb1[i] * zx1[j]) + (zb2[i] * mean(zx2)) + (zb4[i] * mean(zx4))
  }
}
```

Spending: minimums
Based on model3 (salary: zx3 and b3 removed)

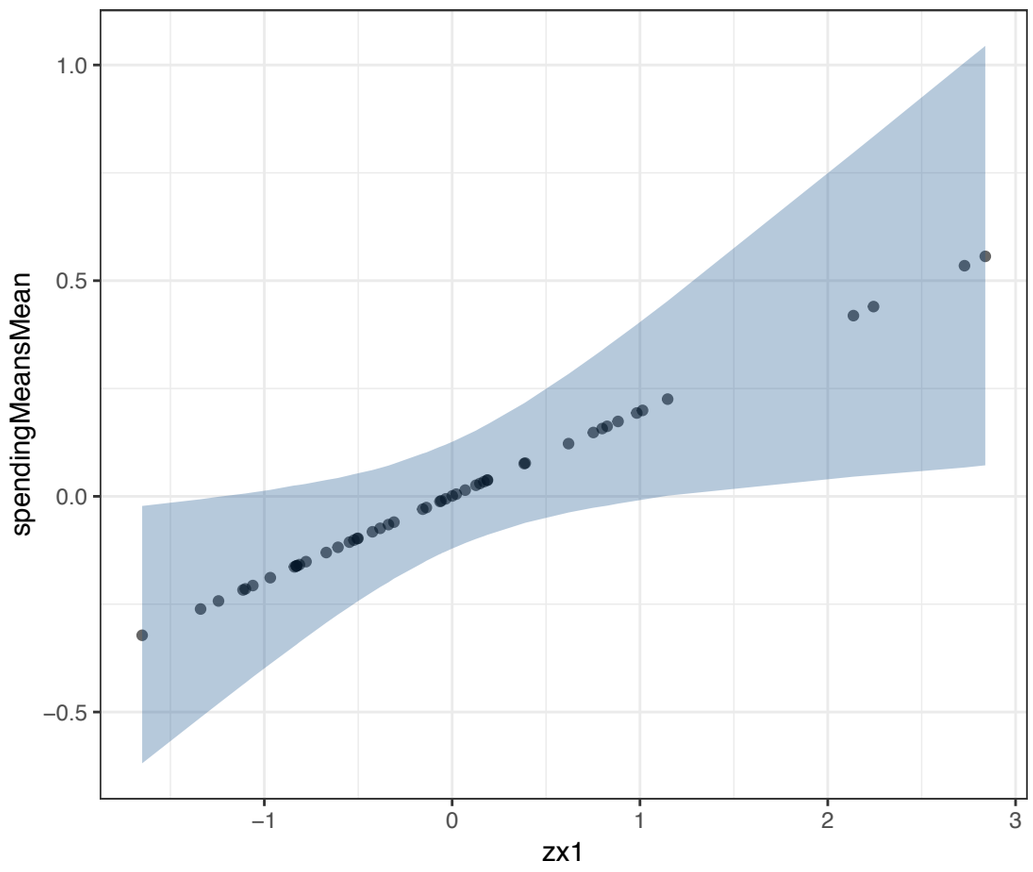
- Calculate mean predicted values, and associated HDI

```
spendingMeansMean = apply(spendingMeans, 2, mean)
spendingMeansLow = apply(spendingMeans, 2, quantile, probs = 0.025)
spendingMeansHigh = apply(spendingMeans, 2, quantile, probs = 0.975)
```

Spending: means
Based on model3 (salary: zx3 and b3 removed)

- Plot predictions

```
spendingMeansSummary = data.frame(zx1, spendingMeansMean, spendingMeansLow, spendingMeansHigh)

ggplot(spendingMeansSummary) +
  theme_bw() +
  geom_point(aes(x = zx1, y = spendingMeansMean), alpha = 0.6) +
  geom_ribbon(aes(x = zx1, ymin = spendingMeansLow, ymax = spendingMeansHigh), fill =
    “dodgerblue4", alpha = 0.3)
```



Spending: max
Based on model3 (salary: zx3 and b3 removed)

- Calculate predicted `y` values, given the **maximum** values for all other parameters

```
spendingMax = matrix(0, nrow = length(zb0), ncol = length(zy))

for (i in 1:length(zb0)) {
  for (j in 1:length(zy)) {
    spendingMax[i, j] = zb0[i] + (zb1[i] * zx1[j]) + (zb2[i] * max(zx2)) + (zb4[i] * max(zx4))
  }
}
```

Spending: max

Based on model3 (salary: zx3 and b3 removed)

- Calculate predicted y val es for all other parameters

`zx1` will vary across all observed values

```
spendingMax = matrix(0, nrow = length(zb0), ncol = length(zy))

for (i in 1:length(zb0)) {
  for (j in 1:length(zy)) {
    spendingMax[i, j] = zb0[i] + (zb1[i] * zx1[j]) + (zb2[i] * max(zx2)) + (zb4[i] * max(zx4))
  }
}
```

Spending: max

Based on model3 (salary: zx3 and b3 removed)

- Calculate predi **num** values for all other parameters

coefficient for other variables multiplied by the **max** value for each

```
spendingMax = matrix(0, nrow = length(zb0), ncol = length(zy))

for (i in 1:length(zb0)) {
  for (j in 1:length(zy)) {
    spendingMax[i, j] = zb0[i] + (zb1[i] * zx1[j]) + (zb2[i] * max(zx2)) + (zb4[i] * max(zx4))
  }
}
```

Spending: max

Based on model3 (salary: zx3 and b3 removed)

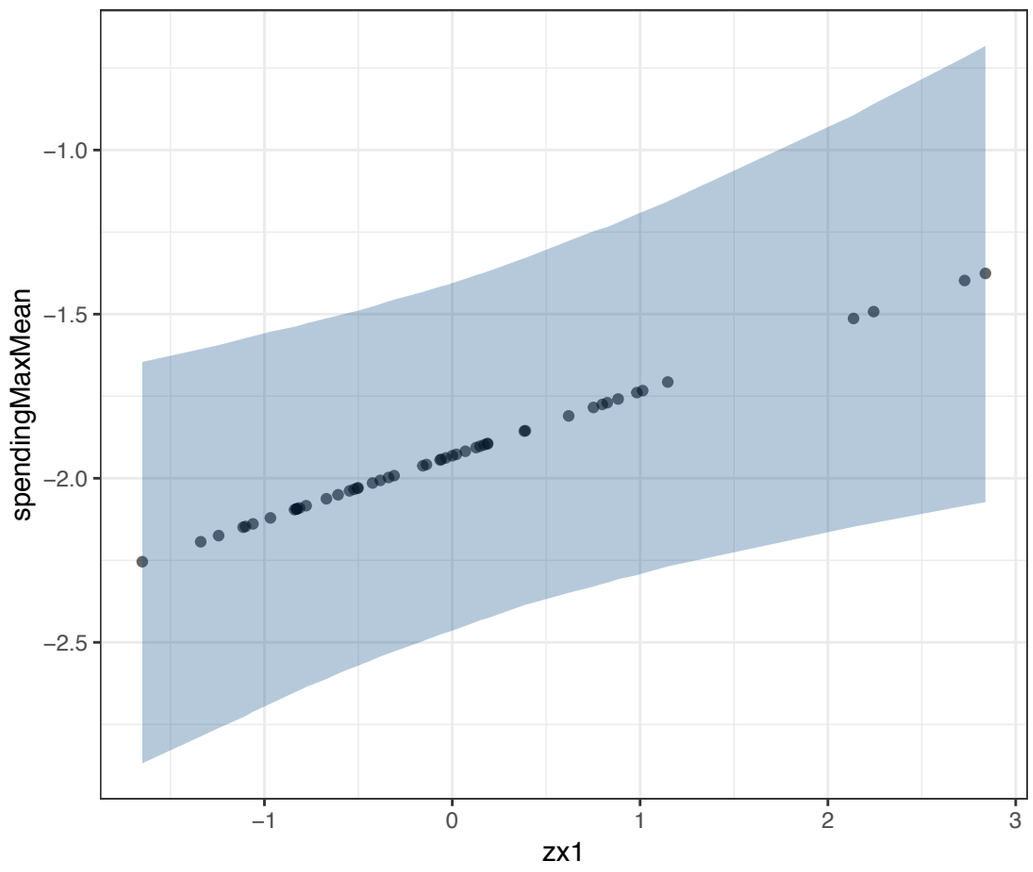
- Calculate mean predicted values, and associated HDI

```
spendingMaxMean = apply(spendingMax, 2, mean)
spendingMaxLow = apply(spendingMax, 2, quantile, probs = 0.025)
spendingMaxHigh = apply(spendingMax, 2, quantile, probs = 0.975)
```

Spending: max
Based on model3 (salary: zx3 and b3 removed)

- Plot predictions

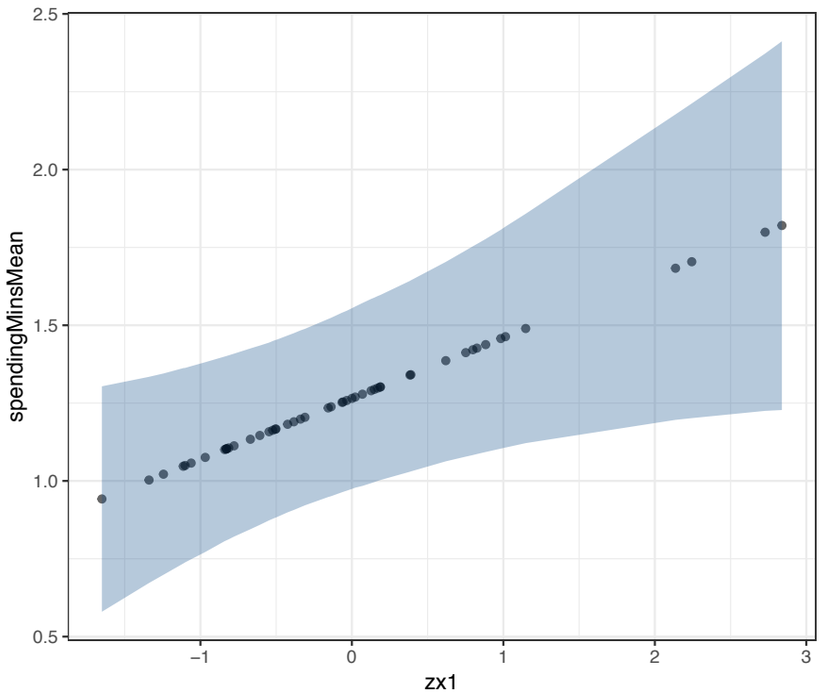
```
spendingMaxSummary = data.frame(zx1, spendingMaxMean, spendingMaxLow, spendingMaxHigh)

ggplot(spendingMaxSummary) +
  theme_bw() +
  geom_point(aes(x = zx1, y = spendingMaxMean), alpha = 0.6) +
  geom_ribbon(aes(x = zx1, ymin = spendingMaxLow, ymax = spendingMaxHigh), fill = "dodgerblue4",
    alpha = 0.3)
```



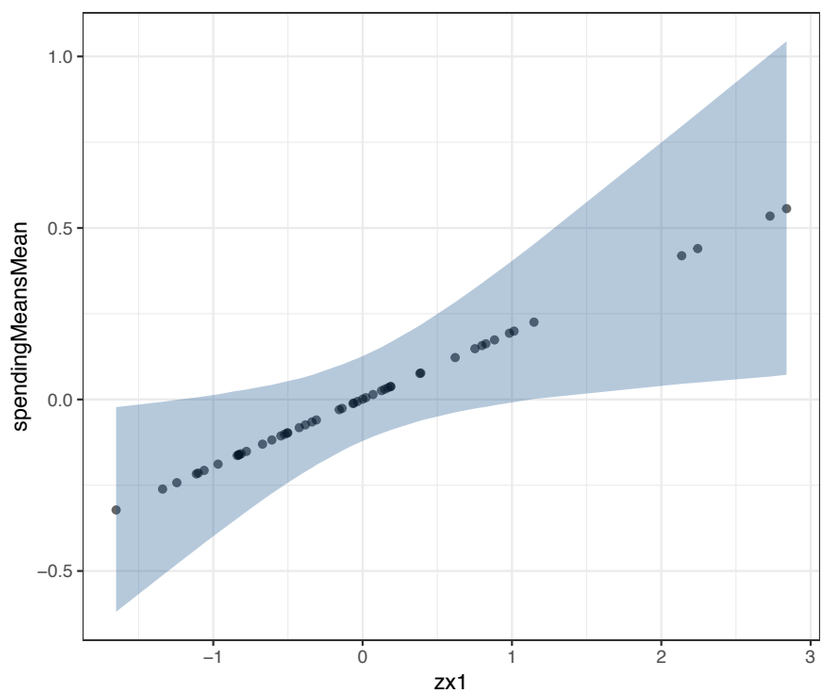
Spending

Based on model3 (salary: zx3 and b3 removed)

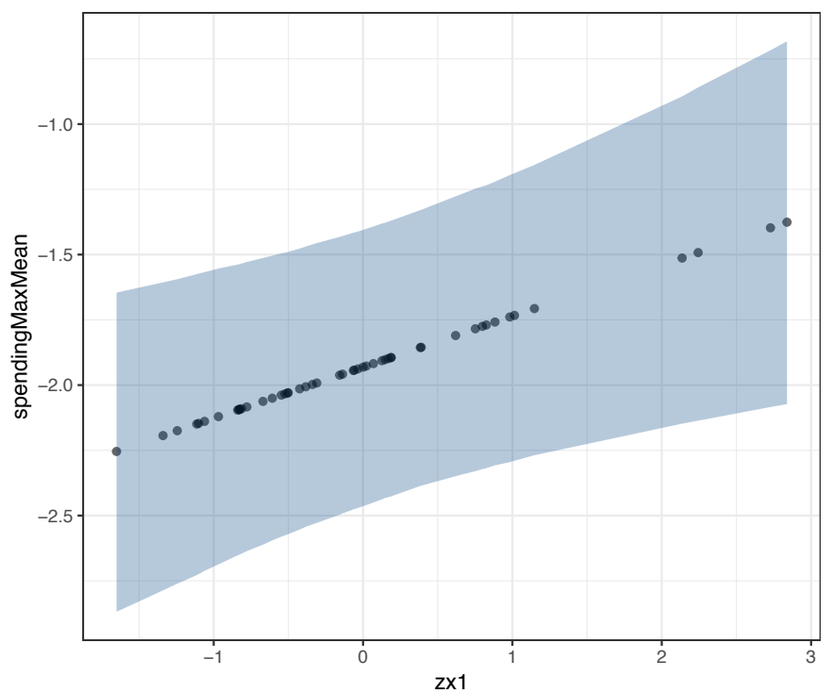
Min



Mean



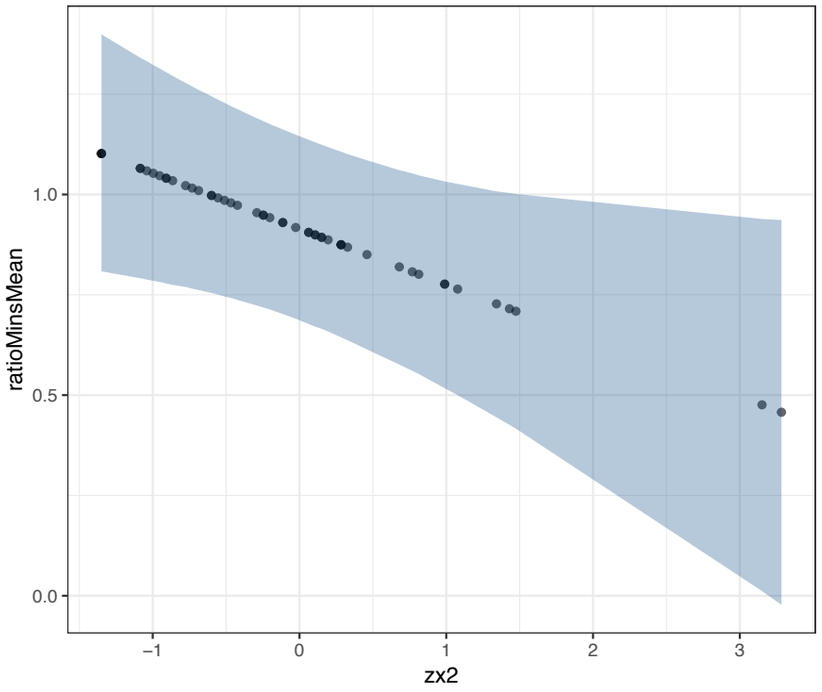
Max



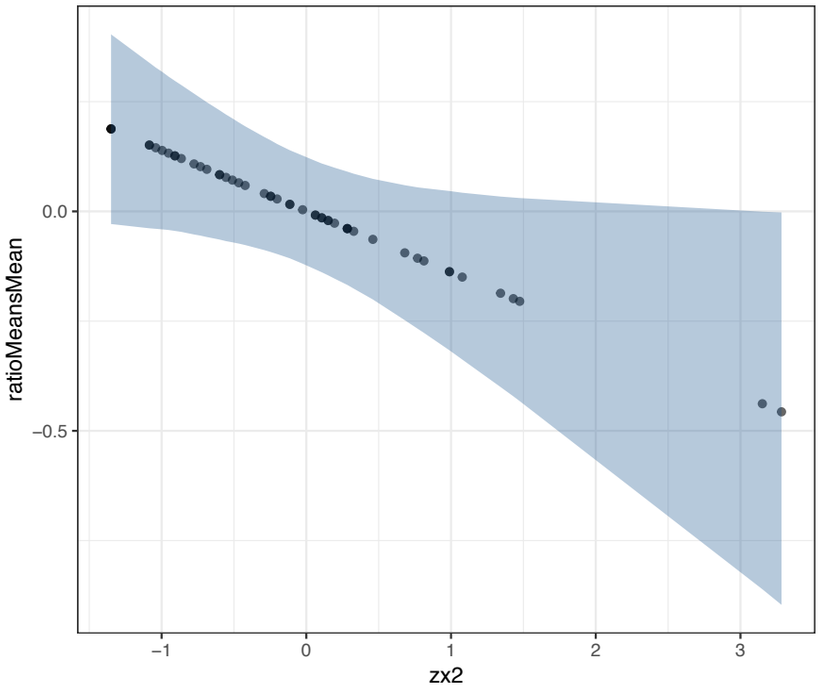
Student:Teacher Ratio

Based on model2 (spend: zx1 and b1 removed)

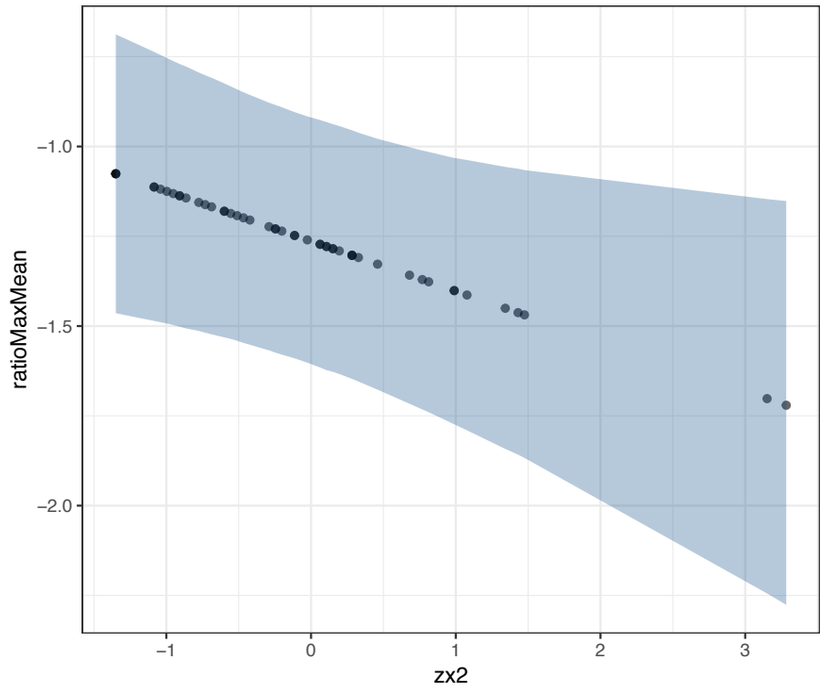
Min



Mean



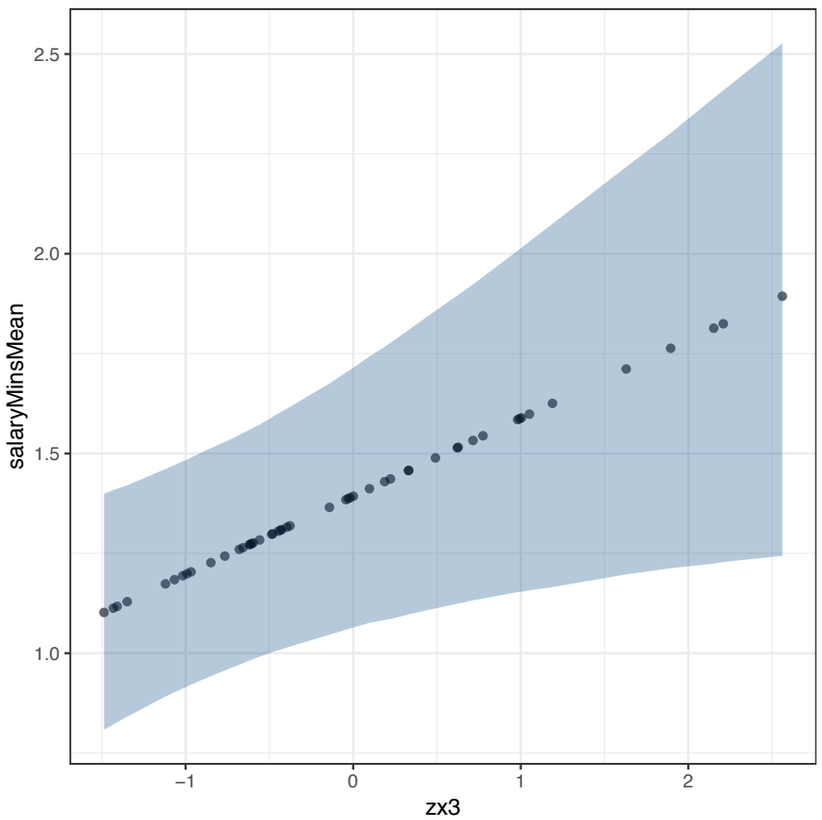
Max



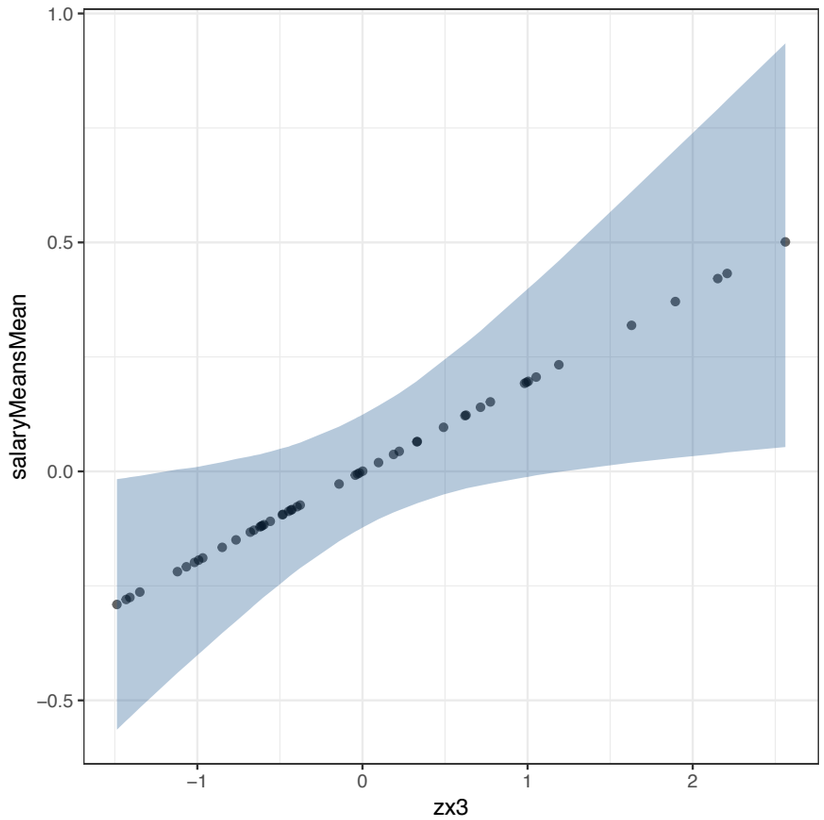
Teacher Salary

Based on model2 (spend: zx1 and b1 removed)

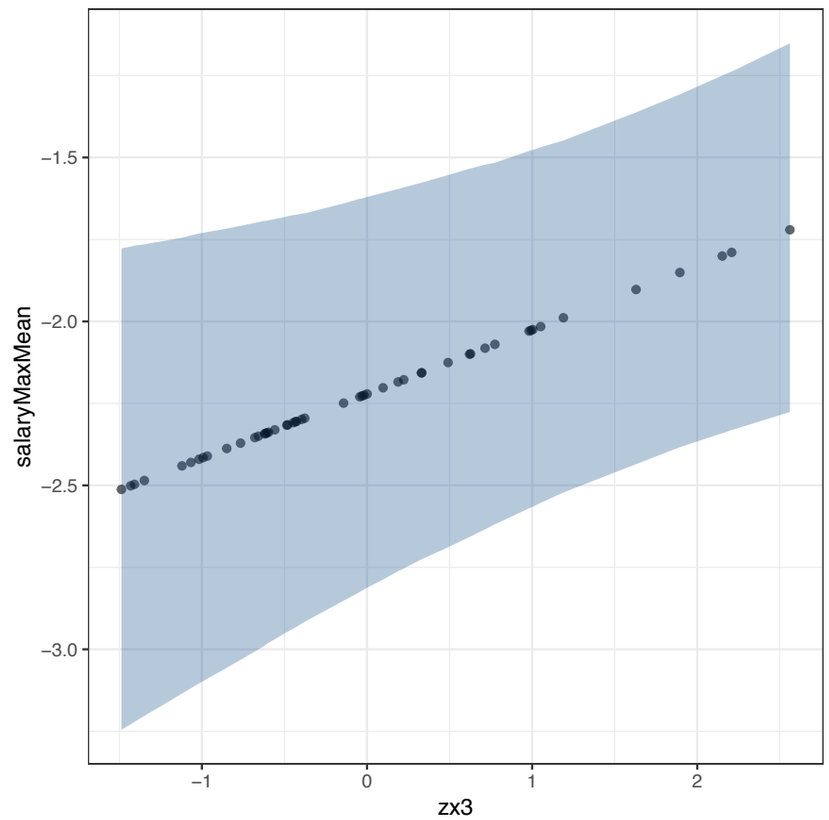
Min



Mean



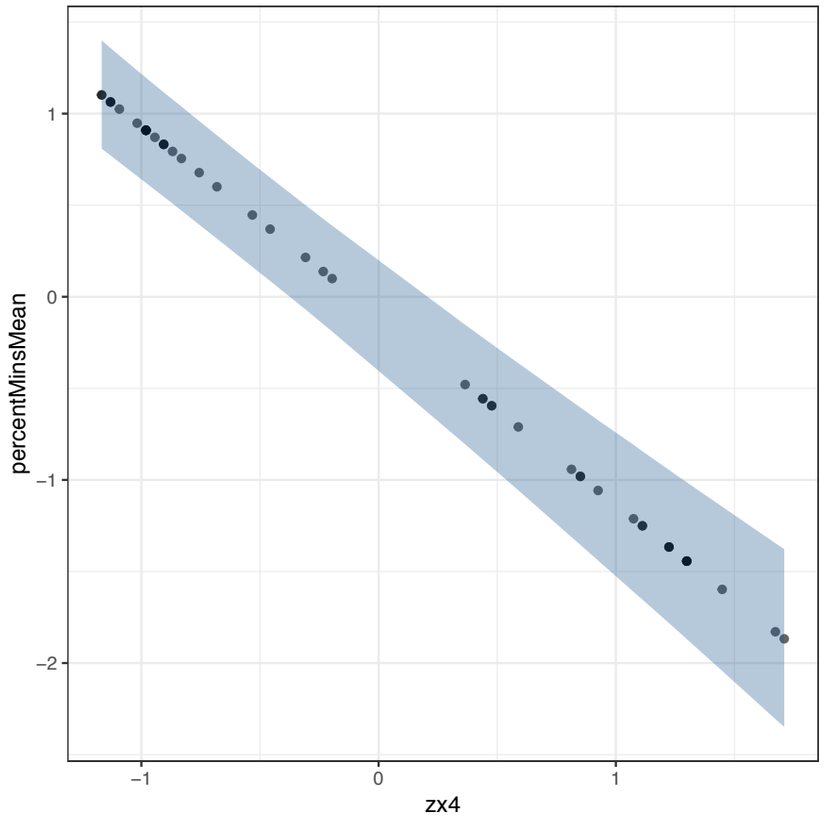
Max



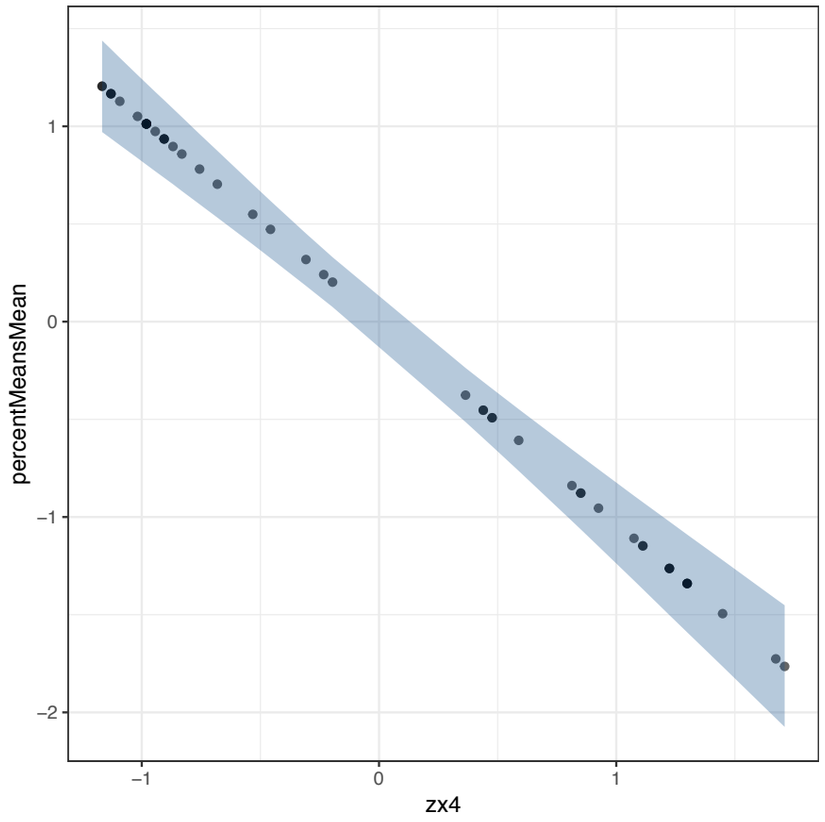
% Students Taking SAT

Based on model2 (spend: zx1 and b1 removed)

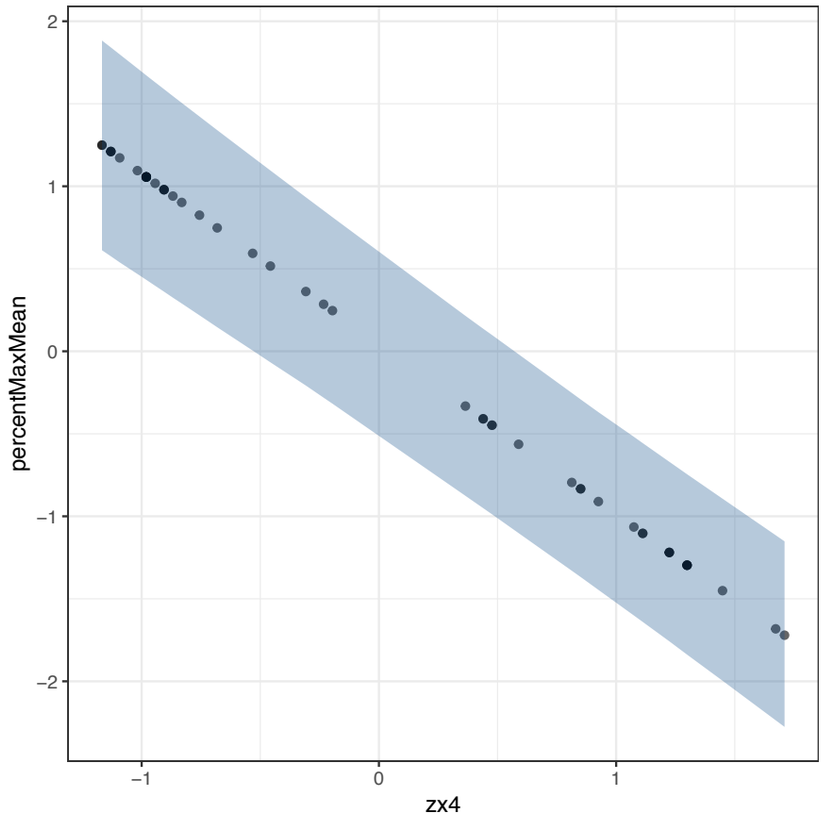
Min



Mean



Max



Conclusions

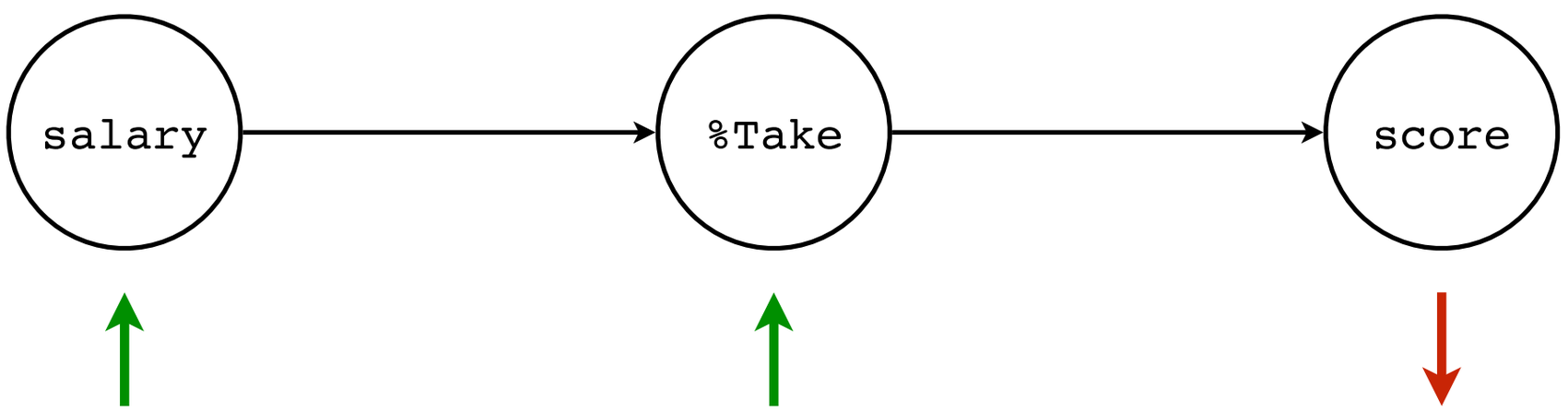
1. All predictor variables appear important
   - Spending and teacher salary redundant, because they are highly correlated

2. Initial plot suggesting negative relationship between teacher salary and SAT scores is erroneous
   - Due to effects of other terms
   - True relationship is positive

3. Percentage of students taking the SAT has the largest effect
   - Effect is negative

Conclusions

- Proposed explanation



Questions?