Linear Programming 1

Solution 1: Sparse Quantile Regression

(a) Univariate sparse quantile regression:

$$\min_{(\beta_0,\beta_1)\in\mathbb{R}^2} \frac{1}{n}\sum_{i=1}^{n} \rho_\tau(y^{(i)} - \beta_0 - \beta_1 x^{(i)}) \text{ s.t. } |\beta_1| \leq t$$

(i) Decompose unconstrained parameters into positive and negative part: $\beta_i = \beta_i^+ - \beta_i^- \quad i = 0, 1$

(ii) Transform absolute value: $|\beta_1| \leq t \iff \beta_1^+ + \beta_1^- \leq t$

(iii) We can write $\rho_\tau(\underbrace{y^{(i)} - \beta_0 - \beta_1 x^{(i)}}_{=:r^{(i)}}) = \tau \cdot r^{(i)} \mathbb{1}_{\{r^{(i)}>0\}} - (1-\tau) \cdot r^{(i)} \mathbb{1}_{\{r^{(i)}\leq 0\}} = \tau \cdot {r^{(i)}}^+ + (1-\tau) \cdot {r^{(i)}}^-$

(iv) Transform equality of the residuals into two inequalities:
${r^{(i)}}^+ - {r^{(i)}}^- = y^{(i)} - \beta_0 - \beta_1 x^{(i)} \iff {r^{(i)}}^+ - {r^{(i)}}^- \leq y^{(i)} - \beta_0 - \beta_1 x^{(i)}$ and
$-{r^{(i)}}^+ + {r^{(i)}}^- \leq -y^{(i)} + \beta_0 + \beta_1 x^{(i)}$

With this we get the standard form:

$$\max_{\mathbf{z}\in\mathbb{R}^{4+2n}} \mathbf{c}^\top \mathbf{z}$$

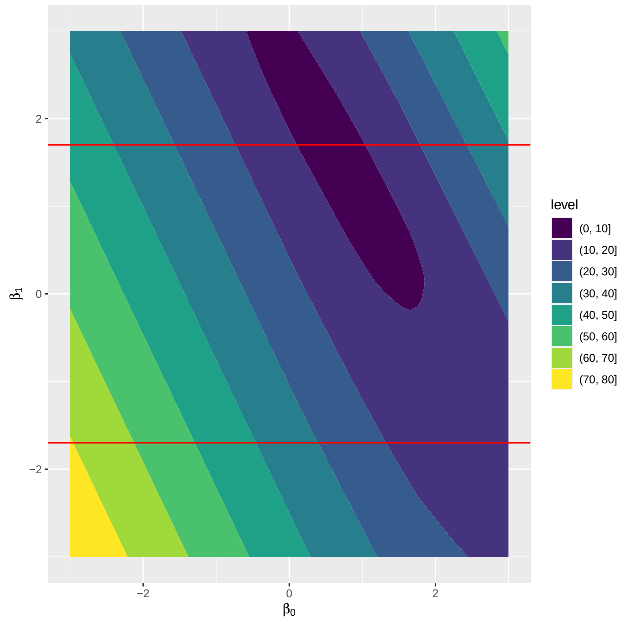
s.t.

$$\mathbf{A}\mathbf{z} \leq \mathbf{b},$$
$$\mathbf{z} \geq \mathbf{0}$$

with $\mathbf{z} = \begin{pmatrix} \beta_0^+ \\ \beta_0^- \\ \beta_1^+ \\ \beta_1^- \\ {r^{(1)}}^+ \\ \vdots \\ {r^{(n)}}^+ \\ {r^{(1)}}^- \\ \vdots \\ {r^{(n)}}^- \end{pmatrix}, \mathbf{c} = \begin{pmatrix} 0 \\ 0 \\ 0 \\ 0 \\ -\tau \\ \vdots \\ -\tau \\ -(1-\tau) \\ \vdots \\ -(1-\tau) \end{pmatrix}, \mathbf{A} = \begin{pmatrix} 0 & 0 & 1 & 1 & 0 & \cdots & 0 \\ \mathbf{1}_n & -\mathbf{1}_n & \mathbf{x} & -\mathbf{x} & \mathbf{I}_n & & -\mathbf{I}_n \\ -\mathbf{1}_n & \mathbf{1}_n & -\mathbf{x} & \mathbf{x} & -\mathbf{I}_n & & \mathbf{I}_n \end{pmatrix}, \mathbf{b} = \begin{pmatrix} t \\ \mathbf{y} \\ -\mathbf{y} \end{pmatrix}$

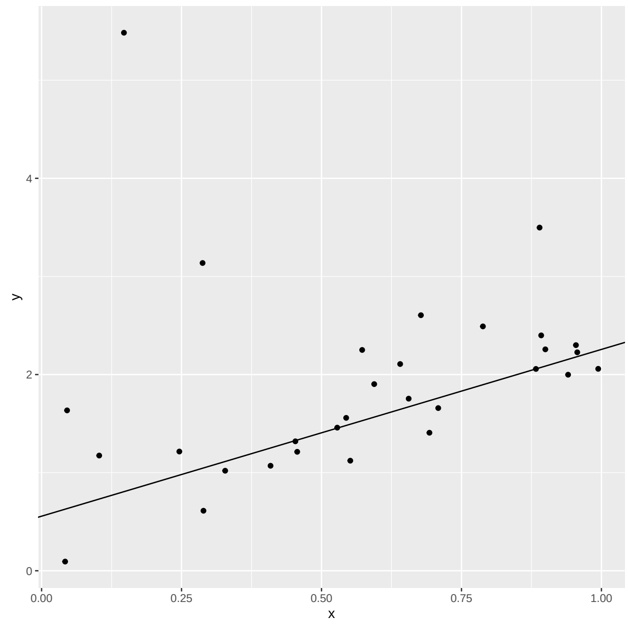
(b)
```r
library(ggplot2)

set.seed(123)
n = 30
x = runif(n)
y = 2 * x + rgamma(n, shape = 1)

remp = function(beta){
 r = y - beta[1] - beta[2] * x
 return(sum(ifelse(r > 0, tau*r, -(1-tau)*r)))
}

tau = 0.4
```

```r
tval = 1.7

b = seq(-3, 3, by=0.05)
bb = expand.grid(X1 = b, X2 = b)
fbb = apply(bb, 1, function(beta) remp(beta))

df = data.frame(bb = bb, fbb = fbb)
remp_plot = ggplot() +
 geom_contour_filled(data = df, aes(x = bb.X1, y = bb.X2, z = fbb)) +
 xlab(expression(beta[0])) +
 ylab(expression(beta[1])) +
 geom_hline(yintercept = tval, color="red") +
 geom_hline(yintercept = -tval, color="red")

remp_plot
```

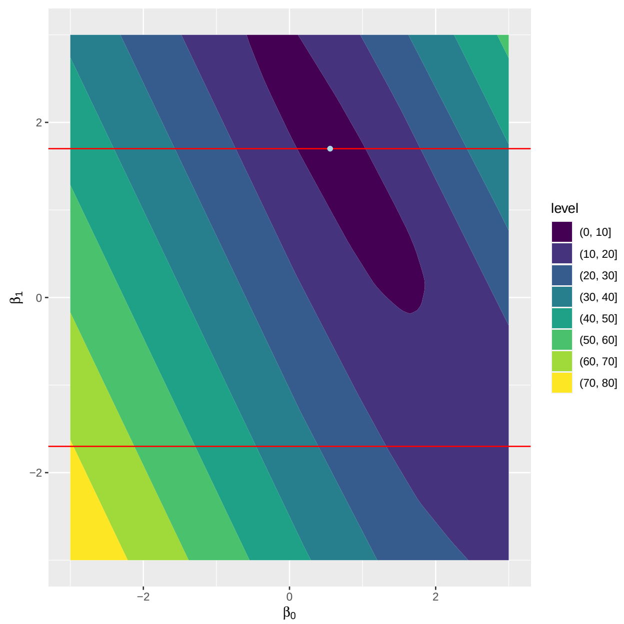


(c)

```r
library(linprog)
## Loading required package:  lpSolve

Amat = c(0, 0, 1, 1, rep(0, 2*n))
Amat = rbind(Amat, cbind(1, -1, x, -x, diag(n), -diag(n)))
Amat = rbind(Amat, cbind(-1, 1, -x, x, -diag(n), diag(n)))

bvec = c(tval, y, -y)
cvec = c(0, 0, 0, 0, rep(tau, n), rep(1-tau, n))

res = solveLP(cvec, bvec, Amat, maximum = FALSE, lpSolve = TRUE)

beta = c(res$solution[1] - res$solution[2], res$solution[3] - res$solution[4])

ggplot(data.frame(x = x, y = y), aes(x=x, y=y)) +
 geom_point() +
 geom_abline(intercept = beta[1], slope = beta[2])
```




```
remp_plot +   geom_point(data=data.frame(x = beta[1], y = beta[2]),
          aes(x=x, y=y), color="lightblue")
```



(d) From the lecture, we know that the dual problem must be

$$\max_{\boldsymbol{\alpha} \in \mathbb{R}^{2n+1}} -\boldsymbol{\alpha}^\top \mathbf{b}$$

s.t.

$$-\boldsymbol{\alpha}^\top \mathbf{A} \leq -\mathbf{c}^\top$$

$$\boldsymbol{\alpha} \geq \mathbf{0}$$