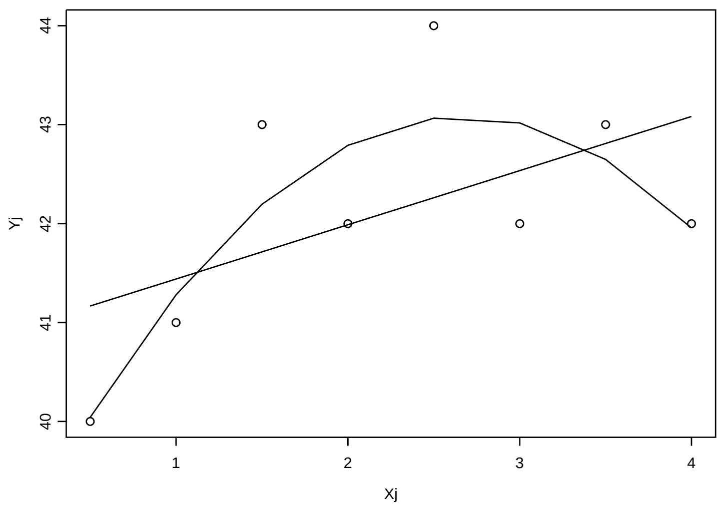

Figure 1: Plot of P602, 6

P602, 2

From equation (10.1.5) we know that

$$\hat{\beta}_1 = \bar{y}_n - \hat{\beta}_2 \bar{x}_n$$

So $(\bar{x}_n, \bar{y}_n)$ is on the line $y = \hat{\beta}_1 + \hat{\beta}_2 x$.

P602, 5

The parabola can get at most the same sum of the squares of deviation as the straight line when we take the coefficient β_3 of x^2 to be 0.

P602, 6

```
> Xj <- seq(0.5, 4.0, length=8)
> Xj
[1] 0.5 1.0 1.5 2.0 2.5 3.0 3.5 4.0
> Yj_c(40, 41, 43, 42, 44, 42, 43, 42)
> plot(Xj, Yj)
> fit_lm(Yj ~ Xj)
> lines(Xj, fit$fitted.value)
> fit2_lm(Yj ~ Xj+Xj^2)
> lines(Xj, fit2$fitted.value)
```

P611, 11

```
> # Vectors x, y with data from DeGroot Table 10.1:
> Xj <- c(1.9,  0.8,  1.1,  0.1, -0.1, 4.4, 4.6, 1.6, 5.5, 3.4);
```

```
> Yj <- c(0.7, -1.0, -0.2, -1.2, -0.1, 3.4, 0.0, 0.8, 3.7, 2.0);
>
> plot(Xj,Yj, xlim=c(-2,7), ylim=c(-2,5));
> fit <- lm(Yj~Xj);
> abline(fit,col=4);
>
> print(summary(fit));
>
> Xj_2
> y_as.data.frame(Xj)
> predict.lm(fit, y, type=c("response"),se.fit=T);
$fit:
         1
 0.5839361

$se.fit:
         1
 0.3474879

$residual.scale: [1] 1.082637

$df: [1] 8

> Xj <- c(1.9,  0.8,  1.1,  0.1, -0.1, 4.4, 4.6, 1.6, 5.5, 3.4);
> Yj <- c(0.7, -1.0, -0.2, -1.2, -0.1, 3.4, 0.0, 0.8, 3.7, 2.0);
> x_2
> n_10
# the coefficient in equation 10.2.17
> (sum((Xj-x)^2))/(n*sum((Xj-mean(Xj))^2))+1
[1] 1.103018
> # sigma^2, equation 10.2.4
> sum((fit$residual)^2)/n
[1] 0.9376817
> 1.103018*0.9376817^2
[1] 0.9698252
```

P626, 2

```
> Xj_c(0.3, 1.4, 1.0, -0.3, -0.2, 1.0, 2.0, -1.0, -0.7, 0.7)
> Yj_c(0.4, 0.9, 0.4, -0.3, 0.3, 0.8, 0.7, -0.4, -0.2, 0.7)
> fit_lm(Yj ~ Xj)
> summary(fit)
> n_10
> # equation 10.3.18
> sqrt(n*(n-2)*sum((Xj-mean(Xj))^2)/sum(Xj^2))
[1] 8.13082
> beta1.hat_fit$coefficient[1]
> beta1.star_0.7
> (beta1.hat-beta1.star)/sqrt(sum(fit$residual^2))
 (Intercept)
  -0.8233579
>  8.13082*-0.8233579
[1] -6.694575
```

```
> 2*pt(-6.694575, fit$df.residual)
[1] 0.0001535629
```

Thus H_0 is rejected.