Part 1a Summary

Our goal is to estimate a Cobb-Douglas production function with a logarithmic transformation of sales, labor and capital. While doing so, we must use time dummies to control for the unobserved effects related to time. First, I made a logarithmic transformation of the sales, labor, and capital variables. When making the OLS model, I included factor(year) to control for the year-specific effects. The OLS model with time dummies looks like $log(sales)_{it} = \alpha_0 + \alpha_L log(labor)_{it} + \alpha_K log(capital)_{it} + \gamma_t + u_{it}$, where α_0 represents the constant term, α_L and α_K are the coefficients for labor and capital respectively, γ_t represents the time dummies, and u_{it} is the error term. I got an estimate of 3.046843 for α_0, 0.557884 for α_L, and 0.432283 for α_K. The Adjusted R-squared for this model is 0.9692.

Part 1a Code

```
library(haven)
library(dplyr)
data <- read_dta("C:/Users/eugen/Downloads/blundell.dta") %>%
  mutate(log_sales = log(sales),
         log_labor = log(labor),
         log_capital = log(capital))
ols_model <- lm(log_sales ~ log_labor + log_capital + factor(year), data = data)
summary(ols_model)
```

```
Call:
lm(formula = log_sales ~ log_labor + log_capital + factor(year),
    data = data)

Residuals:
     Min       1Q   Median       3Q      Max
-1.33046 -0.21402 -0.02862  0.19592  1.58696

Coefficients:
                  Estimate Std. Error t value Pr(>|t|)
(Intercept)       3.046843   0.031527  96.644  < 2e-16 ***
log_labor         0.557884   0.009829  56.761  < 2e-16 ***
log_capital       0.432283   0.008140  53.108  < 2e-16 ***
factor(year)1983 -0.056863   0.022107  -2.572  0.01014 *
factor(year)1984 -0.050041   0.022134  -2.261  0.02382 *
factor(year)1985 -0.087571   0.022198  -3.945 8.12e-05 ***
factor(year)1986 -0.092866   0.022269  -4.170 3.11e-05 ***
factor(year)1987 -0.058093   0.022304  -2.605  0.00923 **
factor(year)1988 -0.021163   0.022328  -0.948  0.34327
factor(year)1989 -0.038292   0.022437  -1.707  0.08796 .
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 0.3526 on 4062 degrees of freedom
Multiple R-squared:  0.9693,    Adjusted R-squared:  0.9692
F-statistic: 1.425e+04 on 9 and 4062 DF,  p-value: < 2.2e-16
```

Part 1b Summary

To test the null hypothesis $\alpha_L + \alpha_K = 1$, I used a linear hypothesis statistical test. The full model that allows α_L and α_K to move freely without restriction is the OLS model from part a. To test the restriction of our null hypothesis, I used linearHypothesis(), which takes the estimated full model and specifically tests whether $\alpha_L + \alpha_K = 1$ holds true. This is similar to the Wald Test, however here it isn't necessary to specify a restricted model; linearHypothesis() does it for us. From this hypothesis test, the p-value is 0.002314, and it is the key value that is used to determine whether to reject or accept the null. As it is significantly smaller than 0.05, the null hypothesis is rejected at the 5% significance level. Since the sum of the coefficients is significantly different from 1, it suggests that the returns to scale are not constant. It would fit our dataset better to use the model that allows freely estimated coefficients for log(labor) and log(capital).

Part 1b Code

```
library(lmtest)
library(car)
ols_model <- lm(log_sales ~ log_labor + log_capital + factor(year), data = data)
linearHypothesis(ols_model, "log_labor + log_capital = 1")

Linear hypothesis test

Hypothesis:
log_labor  + log_capital = 1

Model 1: restricted model
Model 2: log_sales ~ log_labor + log_capital + factor(year)

  Res.Df    RSS Df Sum of Sq      F   Pr(>F)
1   4063 506.05
2   4062 504.90  1    1.1552 9.2938 0.002314 **
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Part 2a Summary

To estimate a fixed effects model, firm-specific effects must be controlled to account for the unobserved heterogeneity. Each firm will have its own intercept, allowing the model to capture individual characteristics that don't vary over time. The time dummies take account for the year-specific effects. The Fixed Effects model including the time dummies, where η_i are the individual-specific fixed effects for each firm, is $log(sales)_{it} = \alpha_0 + \alpha_L log(labor)_{it} + \alpha_K log(capital)_{it} + \gamma_t + \eta_i + u_{it}$. The other variables remain the same from question one. Using plm(), the model is created with factor(year) representing the time dummies. The model = "within" input indicates a Fixed Effects model. This will estimate separate intercept for each firm, and will capture individual heterogeneity. I got an estimate of 0.6544609 for α_L, and 0.2329072 for α_K. The Adjusted R-squared for this model is 0.69972. The estimates for each year are shown in the summary below.

Part 2a Code

```
library(plm)
fe_model <- plm(log_sales ~ log_labor + log_capital + factor(year),
            data = data,
            model = "within")
summary(fe_model)
```

```
Oneway (individual) effect Within Model

Call:
plm(formula = log_sales ~ log_labor + log_capital + factor(year),
    data = data, model = "within")

Balanced Panel: n = 509, T = 8, N = 4072

Residuals:
       Min.     1st Qu.      Median     3rd Qu.        Max.
-0.93284482 -0.06677120  0.00040555  0.06495062  1.26649972

Coefficients:
                   Estimate Std. Error t-value  Pr(>|t|)
log_labor          0.6544609  0.0144048 45.4336 < 2.2e-16 ***
log_capital        0.2329072  0.0136370 17.0791 < 2.2e-16 ***
factor(year)1983 -0.0376406  0.0093042 -4.0455 5.331e-05 ***
factor(year)1984 -0.0076445  0.0096071 -0.7957  0.426250
factor(year)1985 -0.0234513  0.0100955 -2.3229  0.020238 *
factor(year)1986 -0.0136103  0.0105543 -1.2895  0.197294
factor(year)1987  0.0314121  0.0108748  2.8885  0.003894 **
factor(year)1988  0.0753576  0.0111072  6.7846 1.358e-11 ***
factor(year)1989  0.0764165  0.0118166  6.4669 1.137e-10 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Total Sum of Squares:    293.57
Residual Sum of Squares: 76.959
R-Squared:      0.73785
Adj. R-Squared: 0.69972
F-statistic: 1111.47 on 9 and 3554 DF, p-value: < 2.22e-16
```

Part 2b Summary

The null hypothesis to be tested is $\eta_i = 0$ for every firm i, which means that there are no unobserved individual-specific effects and that the Fixed Effects model is not necessary for this dataset. Instead, it's possible that a Pooled model is more fitting, so it is crucial to test this hypothesis by comparing the two models using a F test. The Fixed Effects model was estimated in part a, so the next step is to estimate the Pooled model. The Pooled model, which is without fixed effects, treats all firms as if they are homogeneous, and it doesn't account for individual firm effects. Once both models are estimated and summarized, the F-test is used to compare the two models, and it will help determine if the firm fixed effects are significant. The pFtest() function gives us the F-statistic for comparing the two models and the p-value, which will ultimately show whether the firm fixed effects are significant. The F-statistic was 38.903, which already indicates that including individual firm effects would significantly improve the model.

Additionally, the p-value is less than 2.2e-16 and is extremely small, being much smaller than 0.05. The null hypothesis that $\eta_i = 0$ is rejected, and it is concluded that the Fixed Effects model is most appropriate, not the Pooled model. This strongly suggests that there is significant time-invariant unobserved heterogeneity across firms.

Part 2b Code

```
fe_model <- plm(log_sales ~ log_labor + log_capital + factor(year),
            data = data, model = "within")
pooled_model <- plm(log_sales ~ log_labor + log_capital + factor(year),
                data = data, model = "pooling")
fe_Ftest <- pFtest(fe_model, pooled_model)
print(fe_Ftest)
```

```
        F test for individual effects

data:  log_sales ~ log_labor + log_capital + factor(year)
F = 38.903, df1 = 508, df2 = 3554, p-value < 2.2e-16
alternative hypothesis: significant effects
```

Part 3a Summary

The Arellano-Bond system estimator is a Generalized Method of Moments estimator that is used for dynamic panel data models. This estimator is used to model the relationship between sales, labor, and capital over time for a set of firms, and it uses lagged levels as instruments for the first-differenced equation. This means that the current sales depend on past sales. This is a fitting model to use because we want to model dynamic relationships, where current values rely on the past values. This estimator addresses the possibility of endogeneity by using lagged levels of the dependent variable as instruments for the first-differenced equation, which means that it's using past values of the variables to correct for potential bias in the estimates. In this part, it is assumed that the random error terms (the transitory shocks) are not serially correlated. The shocks are assumed to be independent throughout time. Lagged sales has a coefficient of 0.3375, which is statistically significant since the p-value of 3.585517e-08 is extremely small. This demonstrates that past sales have a strong impact on current sales, and its positive sign indicates that higher sales in the previous period are associated with higher sales in the current period. Labor has a coefficient of 0.3945511, and is statistically significant with an extremely small p-value of 0. Thus, more labor input leads to higher sales. Capital has a coefficient of 0.2492954, and similarly has an extremely small p-value of 2.522427e-13. Like labor, increasing the usage of capital leads to higher sales. Furthermore, we can reject the null hypothesis that the coefficients are zero for all variables.

Part 3a Code

```
arellano_bond <- pgmm(
```

```
  log_sales ~ lag(log_sales, 1) + log_labor + log_capital |
    lag(log_sales, 2),
  data = data,
  index = c("id", "year"),
  effect = "individual",
  model = "twosteps",
  transformation = "d")
coefficients <- coef(arellano_bond)
vcov_matrix <- vcov(arellano_bond)
standard_errors <- sqrt(diag(vcov_matrix))
z_values <- coefficients / standard_errors
p_values <- 2 * (1 - pnorm(abs(z_values)))
ab_summary <- data.frame(
  Coefficients = coefficients,
  Std.Error = standard_errors,
  Z.Value = z_values, P.Value = p_values)
print(ab_summary)
```

```
                  Coefficients  Std.Error   Z.Value      P.Value
lag(log_sales, 1)    0.3375066 0.06125194  5.510138 3.585517e-08
log_labor            0.3945511 0.02922568 13.500151 0.000000e+00
log_capital          0.2492954 0.03406721  7.317752 2.522427e-13
```

Part 3b Summary

Similarly to the first part, an Arellano-Bond system estimator with time dummies is used. However, here it is assumed that there is an AR(1) transitory shock. This model will handle endogeneity because it uses lagged values of endogenous variables as instruments. The AR(1) transitory shock is included because of the assumption that errors are correlated within a firm across adjacent time periods, but otherwise are uncorrelated. Lagged sales has a coefficient of 0.98339273, which is statistically significant since the p-value of 0 is small. This means that past sales performance strongly influences current sales, and its positive sign indicates that higher sales in the previous period are associated with higher sales in the current period. Labor has a coefficient of -0.01821976, and is statistically significant with a small p-value of 0.001895542. The negative coefficient indicates a slightly negative relationship between labor and sales. It may be because there is possible collinearity with capital, or that the firm is over-hiring, but it is uncertain where the negative relationship could stem from. Capital has a coefficient of 0.0.03066815, but it is not statistcally significant due to a p-value of 0.181895007, which is greater than 0.05. This suggests that capital doesn't have a clear, strong impact on sales, which also may be due to possible collinearity with labor. The strongest result that we found was that the large, statistically significant coefficient for the lagged sales implies that firms with strong past performance are very likely to improve their sales. Since a Arellano-Bond system estimator with time dummies and AR(1) shocks was used, the results are less likely to be biased by unobserved time-specific or firm-level effects.

Part 3b Code

```
arellano_bond_ar1 <- pgmm(
  log_sales ~ lag(log_sales, 1) + log_labor + log_capital |
    lag(log_sales, 2:3),
  data = data,
  effect = "individual",
  model = "twosteps",
  transformation = "ld")
ab_ar1_summary <- data.frame(
  Coefficients = coef(arellano_bond_ar1),
  Std.Error = sqrt(diag(vcov(arellano_bond_ar1))),
  Z.Value = (coef(arellano_bond_ar1))/(sqrt(diag(vcov(arellano_bond_ar1)))),
  P.Value = 2 * (1 -
pnorm(abs((coef(arellano_bond_ar1))/(sqrt(diag(vcov(arellano_bond_ar1))))))))
print(ab_ar1_summary)
```

	Coefficients	Std.Error	Z.Value	P.Value
lag(log_sales, 1)	0.98339273	0.019312674	50.919554	0.000000000
log_labor	-0.01821976	0.005865744	-3.106129	0.001895542
log_capital	0.03066815	0.022973379	1.334943	0.181895007

Part 4a Summary

Based on the previous results, my preferred estimate of the production function is the Arellano-Bond Estimator Model. This estimator shows the dynamic nature of the production because through lagged dependent variables, it captures the persistence of sales over time. Additionally, by using these lagged instruments, the estimator provides more accurate and consistent parameter estimates. The structure of this model is very suitable for panel data that has time-specific and firm-level effects. When deciding specifically which Arellano-Bond model is better, I would choose the one that is without AR(1) shocks. Without AR(1) shocks, the estimator still effectively addresses the issue of endogeneity and dynamics, while avoiding potentially overfitting the model. This model gets rid of the unnecessary complexity that adding the shocks creates. Additionally, all of the coefficients were positive and statistically significant, which aligns with what is expected from economic theory. If correlation in shocks is strongly supported by our data, then the Arellano-Bond model with the AR(1) transitory shock may be better.

Part 4b Summary

Looking at the Median TFP line, it's clear that the marginal cost for firms gradually increases as output increases. So, at moderate levels of productivity, firms will experience rising marginal costs when increasing production. Firms with 5th percentile TFP have significantly higher marginal costs at all output levels. These firms are much less efficient in scaling production, which can be due to inefficiencies in input utilization. High-TFP firms (the 95th percentile) have a very flat marginal cost curve. This reflects their high productivity and efficiency, and indicates a great ability to scale production while keeping their costs relatively low. Higher TFP firms benefit from economies of scale. From all the lines, we can see a clear inverse relationship between TFP and marginal costs.

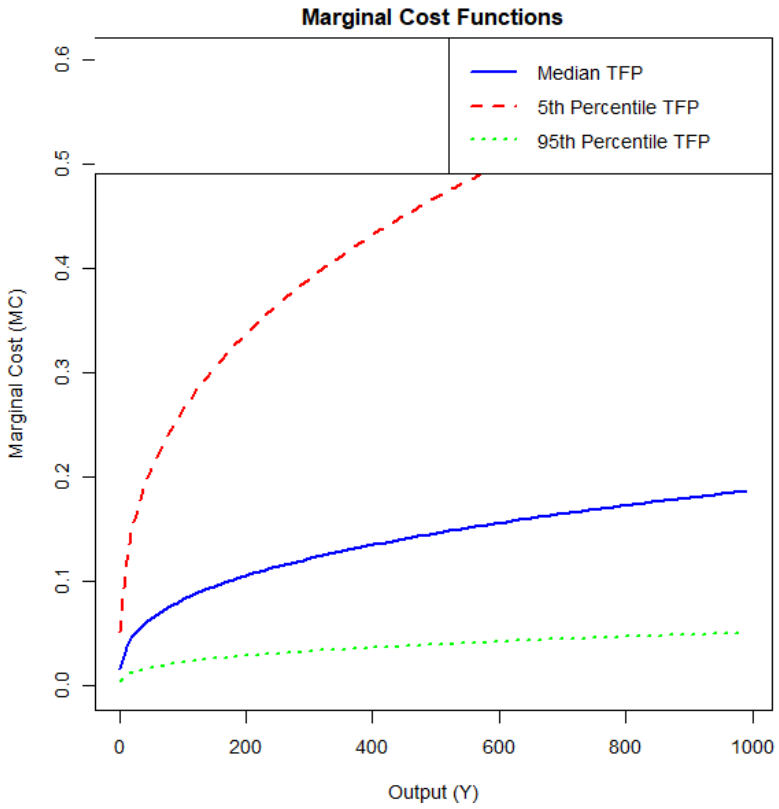
Part 4b Code

```
data <- data %>%
  mutate(log_TFP = log(sales) - (coefficients["log_labor"] * log(labor)) -
           (coefficients["log_capital"] * log(capital))
         )
data_1989 <- subset(data, year == 1989)
tfp_median <- median(data_1989$log_TFP, na.rm = TRUE)
tfp_5th <- quantile(data_1989$log_TFP, 0.05, na.rm = TRUE)
tfp_95th <- quantile(data_1989$log_TFP, 0.95, na.rm = TRUE)

scaled_median <- exp(tfp_median)
scaled_5th <- exp(tfp_5th)
scaled_95th <- exp(tfp_95th)

marginal_cost <- function(Y, A,
alpha_L, alpha_K) {
  (1^alpha_L * 1^alpha_K) / (A *
Y^(alpha_L + alpha_K - 1))
}
Y_values <- seq(1, 1000, by =
10)

MC_median <-
marginal_cost(Y_values,
scaled_median,
coefficients["log_labor"],
coefficients["log_capital"])
MC_5th <-
marginal_cost(Y_values,
scaled_5th,
coefficients["log_labor"],
coefficients["log_capital"])
MC_95th <-
marginal_cost(Y_values,
scaled_95th,
coefficients["log_labor"],
coefficients["log_capital"])
```




```
par(mar = c(4, 4, 2, 1))
plot(Y_values, MC_median, type = "l", col = "blue", lwd = 2, ylim = c(0, max(MC_5th,
MC_95th)),
     xlab = "Output (Y)", ylab = "Marginal Cost (MC)", main = "Marginal Cost Functions")
lines(Y_values, MC_5th, col = "red", lwd = 2, lty = 2)
lines(Y_values, MC_95th, col = "green", lwd = 2, lty = 3)
legend("topright", legend = c("Median TFP", "5th Percentile TFP", "95th Percentile TFP"),
       col = c("blue", "red", "green"), lwd = 2, lty = 1:3)
png("marginal_cost_plot.png", width = 800, height = 600)
dev.off()
```