```
> # ---------------------------------------------------------------------------
> #                       Program Description
> # --------------------- .... [TRUNCATED]

> # Load packages
> library(foreign)

> library(data.table)

> library(AER)

> library(scales)

> library(grid)

> # ===========================================================================
> #                   1. Intensity and Firm Size
> # ================= .... [TRUNCATED]

> # -------------------- Data Processing -------------------------
> # Aggregate ownership rights type
> # 0: missing, 1: State/collective, 3: privat .... [TRUNCATED]

> # State and collective
> sel <- which(KEYFIRM$type == 110 | KEYFIRM$type == 141
+     | KEYFIRM$type == 151 | KEYFIRM$type == 120 | KEYFIRM$type ==  ....
[TRUNCATED]

> KEYFIRM$type_a[sel] <- 1

> # Private
> sel <- which(KEYFIRM$type == 170 | KEYFIRM$type == 171
+     | KEYFIRM$type == 172 | KEYFIRM$type == 173
+     | KEYFIRM$type == 174 | K .... [TRUNCATED]

> KEYFIRM$type_a[sel] <- 3

> # Hong Kong, Macau and Taiwan
> sel <- which(KEYFIRM$type == 200 | KEYFIRM$type == 210
+     | KEYFIRM$type == 220 | KEYFIRM$type == 230 | KEYFIRM$t .... [TRUNCATED]

> KEYFIRM$type_a[sel] <- 4

> # Foreign
> sel <- which(KEYFIRM$type == 300 | KEYFIRM$type == 310
+     | KEYFIRM$type == 320 | KEYFIRM$type == 330 | KEYFIRM$type == 340)

> KEYFIRM$type_a[sel] <- 5
```

```
> # Aggregate treatment technology type
> KEYFIRM$dm1_code_a <- 0

> # Physical
> sel <- which(KEYFIRM$dm1_code >= 1000 & KEYFIRM$dm1_code < 2000)

> KEYFIRM$dm1_code_a[sel] <- 1

> # Chemical
> sel <- which(KEYFIRM$dm1_code >= 2000 & KEYFIRM$dm1_code < 3000)

> KEYFIRM$dm1_code_a[sel] <- 2

> # Physiochemical
> sel <- which(KEYFIRM$dm1_code >= 3000 & KEYFIRM$dm1_code < 4000)

> KEYFIRM$dm1_code_a[sel] <- 3

> # Biological
> sel <- which(KEYFIRM$dm1_code >= 4000 & KEYFIRM$dm1_code < 5000)

> KEYFIRM$dm1_code_a[sel] <- 4

> # Combination
> sel <- which(KEYFIRM$dm1_code >= 5000 & KEYFIRM$dm1_code < 6000)

> KEYFIRM$dm1_code_a[sel] <- 5

> # dm1_code_a == 0 means the equipment is unclassified
>
> KEYFIRM$province <- factor(KEYFIRM$province)

> KEYFIRM$industry_a <- factor(KEYFIRM$industry_a)

> KEYFIRM$Census_Type <- factor(KEYFIRM$Census_Type)

> KEYFIRM$dm1_code_a <- factor(KEYFIRM$dm1_code_a)

> KEYFIRM$type_a <- factor(KEYFIRM$type_a)

> POL5 <- KEYFIRM[industry_a == 22 | industry_a == 13 | industry_a == 15
+     | industry_a == 17 | industry_a == 26]

> POL5 <- POL5[product > 0 & cod_e > 0 & type_a != 0]

> POL5$intensity <- with(POL5, intensity <- cod_e/product)

> # ==================== Regression 1 ========================================
> lm_pol5_all <- lm(log(cod_e) ~ log(product) + province + type_a
+    .... [TRUNCATED]
```

```
> summary(lm_pol5_all)

Call:
lm(formula = log(cod_e) ~ log(product) + province + type_a +
    industry_a, data = POL5)

Residuals:
     Min       1Q   Median       3Q      Max
-13.6730  -1.3044   0.1236   1.4294   7.0787

Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept)  -3.74667    0.17009 -22.028  < 2e-16 ***
log(product)  0.63167    0.00650  97.183  < 2e-16 ***
province12    0.91936    0.20300   4.529 5.95e-06 ***
province13    2.20219    0.17029  12.932  < 2e-16 ***
province14    1.92921    0.19939   9.675  < 2e-16 ***
province15    2.32190    0.20945  11.086  < 2e-16 ***
province21    1.04860    0.17693   5.927 3.13e-09 ***
province22    1.16683    0.20672   5.645 1.67e-08 ***
province23    1.14120    0.20289   5.625 1.87e-08 ***
province31    0.15526    0.17847   0.870   0.3843
province32    0.87671    0.16575   5.289 1.24e-07 ***
province33    1.28765    0.16572   7.770 8.10e-15 ***
province34    1.02464    0.18074   5.669 1.45e-08 ***
province35    1.09303    0.17353   6.299 3.04e-10 ***
province36    1.40749    0.18537   7.593 3.22e-14 ***
province37    1.02133    0.16742   6.100 1.07e-09 ***
province41    1.45293    0.17725   8.197 2.56e-16 ***
province42    1.45913    0.18507   7.884 3.27e-15 ***
province43    1.73715    0.17642   9.847  < 2e-16 ***
province44    0.37106    0.16536   2.244   0.0248 *
province45    2.37634    0.17445  13.622  < 2e-16 ***
province46    1.35907    0.21786   6.238 4.48e-10 ***
province50    0.82452    0.18430   4.474 7.71e-06 ***
province51    1.11428    0.17044   6.538 6.36e-11 ***
province52    1.62967    0.21824   7.467 8.41e-14 ***
province53    1.87311    0.18206  10.288  < 2e-16 ***
province54   -0.82673    1.23624  -0.669   0.5037
province61    1.78717    0.19826   9.014  < 2e-16 ***
province62    2.81659    0.20997  13.414  < 2e-16 ***
province63    2.27521    0.38008   5.986 2.17e-09 ***
province64    3.44462    0.24508  14.055  < 2e-16 ***
province65    1.85097    0.20195   9.165  < 2e-16 ***
type_a3      -0.18557    0.02870  -6.466 1.02e-10 ***
type_a4      -0.46878    0.05507  -8.513  < 2e-16 ***
type_a5      -0.44802    0.04846  -9.245  < 2e-16 ***
industry_a15  0.37125    0.06110   6.076 1.25e-09 ***
industry_a17  0.70557    0.03988  17.695  < 2e-16 ***
```

```
industry_a22  1.34876    0.03972  33.960  < 2e-16 ***
industry_a26 -1.65741    0.03646 -45.454  < 2e-16 ***
---
Signif. codes:  0 ?**?0.001 ?*?0.01 ??0.05 ??0.1 ??1

Residual standard error: 2.121 on 29014 degrees of freedom
Multiple R-squared:  0.4137,    Adjusted R-squared:  0.4129
F-statistic: 538.7 on 38 and 29014 DF,  p-value: < 2.2e-16


> # ==================== Figure 1 ==========================================
> # Residual intensity versus production
> lm_pol5_aux1 <- lm(log(inten .... [TRUNCATED]

> lm_pol5_aux2 <- lm(log(product) ~
+       province + type_a + industry_a, data = POL5)

> POL5$res_intensity <- residuals(lm_pol5_aux1)

> POL5$res_product <- residuals(lm_pol5_aux2)

> pdf("./Results/Figure1.pdf",height=5,width=5)

> plot((POL5$res_intensity)~(POL5$res_product),
+      cex=0.5,mgp=c(1.75, 0.75, 0),
+      xlab="Log Production",ylab="Log Intensity",
+      main=" ..." ... [TRUNCATED]

> pol5_residual <- lm(res_intensity ~ res_product, data = POL5)

> abline(pol5_residual,col="red",lwd=4)

> dev.off()
null device
          1

> # ========================================================================
> #                   2. Firm Size and Technology
> # ================ .... [TRUNCATED]

> clean_share <- (sum(dmtb[5:6]))/sum(dmtb)

> # ==================== Table 2 ===========================================
> # Table 2 Column 2
> PAPER <- KEYFIRM[industry_a == 22]

> dmtb <- table(PAPER$dm1_code_a)

> phyrate <- dmtb[2]/sum(dmtb[2:6])
```

```
> chemrate <- dmtb[3]/sum(dmtb[2:6])

> biorate <- sum(dmtb[5:6])/sum(dmtb[2:6])

> phyrate
        1
0.2588981

> chemrate
        2
0.3364213

> biorate
[1] 0.3859906

> # Table 2 Column 1
> PAPER <- within(PAPER,{
+     cod_eg <- cod_e/cod_g
+     dm1_unit <- dm1_quant/dm1_inv
+     dm1_prod <- dm1_inv/product
+     .... [TRUNCATED]

> phyeffc <- 1 -
+   mean(PAPER$cod_eg[PAPER$dm1_code_a == 1 & PAPER$cod_eg <= 1],na.rm=TRUE)

> chemeffc <- 1 -
+   mean(PAPER$cod_eg[PAPER$dm1_code_a == 2 & PAPER$cod_eg <= 1],na.rm = TRUE)

> bioeffc <- 1 -
+   mean(PAPER$cod_eg[(PAPER$dm1_code_a == 4 | PAPER$dm1_code_a == 5)
+                     & PAPER$cod_eg <= 1],na.rm=TRUE)

> phyeffc
[1] 0.633705

> chemeffc
[1] 0.7496254

> bioeffc
[1] 0.8090434

> # Table 2 Column 3
> median(PAPER$dm1_inv[PAPER$dm1_code_a == 1 & PAPER$cod_eg <= 1],na.rm=TRUE)
[1] 13

> median(PAPER$dm1_inv[PAPER$dm1_code_a == 2 & PAPER$cod_eg <= 1],na.rm=TRUE)
[1] 40

> median(PAPER$dm1_inv[PAPER$dm1_code_a == 4 | PAPER$dm1_code_a == 5
+                      & PAPER$cod_eg <= 1],na.rm=TRUE)
```

```
[1] 120

> # Table 2 Column 4
> median(PAPER$product[PAPER$dm1_code_a == 1 & PAPER$cod_eg <= 1],na.rm=TRUE)
[1] 500

> median(PAPER$product[PAPER$dm1_code_a == 2 & PAPER$cod_eg <= 1],na.rm=TRUE)
[1] 1093

> median(PAPER$product[PAPER$dm1_code_a == 4 | PAPER$dm1_code_a == 5
+                       & PAPER$cod_eg <= 1],na.rm=TRUE)
[1] 2500

> # =========== Unumbered Regression in Section I.B =========================
> # Linear Probability Model of Technology Adoption
> POL5$clean <- 0

> sel <- which(POL5$dm1_code_a == 4 | POL5$dm1_code_a == 5)

> POL5$clean[sel] <- 1

> lm_clean <- lm(clean ~ log(product) + industry_a
+       + province + type_a, data = POL5)

> summary(lm_clean)

Call:
lm(formula = clean ~ log(product) + industry_a + province + type_a,
    data = POL5)

Residuals:
    Min      1Q  Median      3Q     Max
-1.0984 -0.4362  0.1371  0.3836  1.0067

Coefficients:
              Estimate Std. Error t value Pr(>|t|)
(Intercept)   0.226661   0.036117   6.276 3.53e-10 ***
log(product)  0.050771   0.001380  36.785  < 2e-16 ***
industry_a15  0.193266   0.012975  14.895  < 2e-16 ***
industry_a17  0.149922   0.008467  17.706  < 2e-16 ***
industry_a22 -0.171980   0.008433 -20.392  < 2e-16 ***
industry_a26 -0.118606   0.007743 -15.318  < 2e-16 ***
province12    0.009535   0.043105   0.221 0.824936
province13    0.037040   0.036160   1.024 0.305682
province14   -0.142380   0.042341  -3.363 0.000773 ***
province15   -0.324525   0.044476  -7.297 3.02e-13 ***
province21   -0.031092   0.037570  -0.828 0.407920
province22   -0.018096   0.043896  -0.412 0.680154
province23   -0.041018   0.043082  -0.952 0.341062
province31    0.024112   0.037897   0.636 0.524622
```

```
province32    0.052733   0.035196   1.498 0.134072
province33    0.014269   0.035190   0.405 0.685129
province34    0.095733   0.038378   2.494 0.012621 *
province35    0.015461   0.036848   0.420 0.674782
province36   -0.131637   0.039362  -3.344 0.000826 ***
province37    0.095923   0.035550   2.698 0.006975 **
province41    0.098538   0.037638   2.618 0.008848 **
province42   -0.110602   0.039299  -2.814 0.004890 **
province43   -0.200871   0.037461  -5.362 8.29e-08 ***
province44    0.006629   0.035114   0.189 0.850269
province45   -0.224777   0.037043  -6.068 1.31e-09 ***
province46    0.294139   0.046261   6.358 2.07e-10 ***
province50    0.014527   0.039135   0.371 0.710488
province51    0.020746   0.036193   0.573 0.566507
province52   -0.138789   0.046342  -2.995 0.002748 **
province53   -0.132013   0.038660  -3.415 0.000639 ***
province54   -0.260093   0.262509  -0.991 0.321793
province61   -0.081688   0.042100  -1.940 0.052345 .
province62   -0.239881   0.044585  -5.380 7.50e-08 ***
province63   -0.292911   0.080709  -3.629 0.000285 ***
province64   -0.348792   0.052042  -6.702 2.09e-11 ***
province65   -0.277478   0.042883  -6.471 9.92e-11 ***
type_a3       0.006451   0.006094   1.059 0.289796
type_a4       0.037742   0.011694   3.228 0.001250 **
type_a5       0.030796   0.010290   2.993 0.002767 **
---
Signif. codes:  0 ?**?0.001 ?*?0.01 ??0.05 ??0.1 ??1

Residual standard error: 0.4504 on 29014 degrees of freedom
Multiple R-squared:  0.1748,    Adjusted R-squared:  0.1737
F-statistic: 161.7 on 38 and 29014 DF,  p-value: < 2.2e-16


> # ====================== Regressions 2 and 3 ==============================
> # Regression 2
> lm1 <- lm(log(intensity) ~ log(product) + province + .... [TRUNCATED]

> summary(lm1)

Call:
lm(formula = log(intensity) ~ log(product) + province + industry_a +
    type_a, data = POL5[(dm1_code_a == 4 | dm1_code_a == 5) &
    intensity > 0])

Residuals:
    Min      1Q  Median      3Q     Max
-9.7304 -1.2138  0.0905  1.3078  6.7582

Coefficients:
              Estimate Std. Error t value Pr(>|t|)
```

```
(Intercept)  -4.374769   0.201832 -21.675  < 2e-16 ***
log(product) -0.328767   0.008516 -38.605  < 2e-16 ***
province12    0.889520   0.238464   3.730 0.000192 ***
province13    1.957752   0.199367   9.820  < 2e-16 ***
province14    1.507262   0.251767   5.987 2.19e-09 ***
province15    1.334745   0.305127   4.374 1.23e-05 ***
province21    0.837918   0.208160   4.025 5.71e-05 ***
province22    1.065066   0.248824   4.280 1.88e-05 ***
province23    1.046365   0.244212   4.285 1.84e-05 ***
province31    0.060901   0.207311   0.294 0.768941
province32    0.650567   0.192645   3.377 0.000734 ***
province33    0.985663   0.192656   5.116 3.15e-07 ***
province34    0.733646   0.209165   3.507 0.000454 ***
province35    0.823319   0.203296   4.050 5.15e-05 ***
province36    0.991369   0.230861   4.294 1.76e-05 ***
province37    0.984008   0.193573   5.083 3.75e-07 ***
province41    1.364155   0.205159   6.649 3.04e-11 ***
province42    1.006770   0.224382   4.487 7.28e-06 ***
province43    1.253971   0.225356   5.564 2.67e-08 ***
province44    0.095277   0.192749   0.494 0.621096
province45    2.172592   0.219463   9.900  < 2e-16 ***
province46    1.043812   0.240177   4.346 1.39e-05 ***
province50    0.728835   0.219800   3.316 0.000915 ***
province51    0.641214   0.199636   3.212 0.001321 **
province52    0.863803   0.289354   2.985 0.002837 **
province53    1.222854   0.225320   5.427 5.81e-08 ***
province54   -0.265770   1.407365  -0.189 0.850219
province61    1.634067   0.240529   6.794 1.13e-11 ***
province62    1.385544   0.289341   4.789 1.69e-06 ***
province63    1.234785   0.623177   1.981 0.047559 *
province64    2.401801   0.417353   5.755 8.83e-09 ***
province65    1.630515   0.284198   5.737 9.79e-09 ***
industry_a15  0.927166   0.065424  14.172  < 2e-16 ***
industry_a17  1.379724   0.046669  29.564  < 2e-16 ***
industry_a22  1.976992   0.055367  35.707  < 2e-16 ***
industry_a26 -0.911262   0.046754 -19.490  < 2e-16 ***
type_a3      -0.148889   0.036002  -4.136 3.56e-05 ***
type_a4      -0.418407   0.061753  -6.775 1.28e-11 ***
type_a5      -0.494661   0.054677  -9.047  < 2e-16 ***
---
Signif. codes:  0 ?***?0.001 ?**?0.01 ??0.05 ??0.1 ??1

Residual standard error: 1.971 on 16444 degrees of freedom
Multiple R-squared:  0.341,     Adjusted R-squared:  0.3395
F-statistic:   224 on 38 and 16444 DF,  p-value: < 2.2e-16


> # Regression 3
> lm_pool <- lm(log(cod_e) ~ log(product) + province
+     + industry_a + type_a + dm1_code_a,
```

```
+     data = POL5[dm1_code_a == 2 |  .... [TRUNCATED]

> summary(lm_pool)

Call:
lm(formula = log(cod_e) ~ log(product) + province + industry_a + 
    type_a + dm1_code_a, data = POL5[dm1_code_a == 2 | dm1_code_a == 
    1 & intensity > 0])

Residuals:
     Min       1Q   Median       3Q      Max 
-13.5261  -1.3318   0.1716   1.4589   7.0484 

Coefficients:
              Estimate Std. Error t value Pr(>|t|)    
(Intercept)  -3.411377   0.308077 -11.073  < 2e-16 ***
log(product)  0.636428   0.010704  59.455  < 2e-16 ***
province12    0.877321   0.373335   2.350 0.018792 *  
province13    2.398772   0.311135   7.710 1.36e-14 ***
province14    2.305851   0.341491   6.752 1.52e-11 ***
province15    2.499805   0.342710   7.294 3.20e-13 ***
province21    1.634000   0.320426   5.099 3.46e-07 ***
province22    1.369609   0.359516   3.810 0.000140 ***
province23    1.409001   0.357297   3.944 8.08e-05 ***
province31    0.157992   0.329889   0.479 0.632002    
province32    0.972083   0.306410   3.172 0.001515 ** 
province33    1.660023   0.305519   5.433 5.64e-08 ***
province34    1.491665   0.335623   4.444 8.89e-06 ***
province35    1.525967   0.315727   4.833 1.36e-06 ***
province36    1.833069   0.323272   5.670 1.46e-08 ***
province37    1.153179   0.312808   3.687 0.000228 ***
province41    1.491101   0.331443   4.499 6.90e-06 ***
province42    1.983633   0.327685   6.053 1.46e-09 ***
province43    2.128578   0.311388   6.836 8.56e-12 ***
province44    0.717447   0.303951   2.360 0.018272 *  
province45    2.512743   0.309797   8.111 5.53e-16 ***
province46    2.336285   0.474999   4.919 8.84e-07 ***
province50    1.037225   0.328890   3.154 0.001616 ** 
province51    1.694596   0.310655   5.455 5.00e-08 ***
province52    2.055213   0.358467   5.733 1.01e-08 ***
province53    2.171727   0.319901   6.789 1.19e-11 ***
province54   -2.087014   2.237590  -0.933 0.350991    
province61    1.758006   0.348761   5.041 4.71e-07 ***
province62    3.379671   0.345207   9.790  < 2e-16 ***
province63    2.938252   0.533538   5.507 3.73e-08 ***
province64    3.664042   0.370164   9.898  < 2e-16 ***
province65    1.985978   0.334277   5.941 2.91e-09 ***
industry_a15 -0.233624   0.149345  -1.564 0.117770    
industry_a17 -0.345430   0.081089  -4.260 2.06e-05 ***
industry_a22  0.641278   0.063903  10.035  < 2e-16 ***
```

```
industry_a26 -2.498887   0.061544 -40.603  < 2e-16 ***
type_a3      -0.191692   0.046530  -4.120 3.82e-05 ***
type_a4      -0.622441   0.111481  -5.583 2.41e-08 ***
type_a5      -0.305781   0.095740  -3.194 0.001407 **
dm1_code_a2  -0.009086   0.048002  -0.189 0.849879
---
Signif. codes:  0 ?**?0.001 ?*?0.01 ??0.05 ??0.1 ??1

Residual standard error: 2.21 on 11636 degrees of freedom
Multiple R-squared:  0.4143,    Adjusted R-squared:  0.4124
F-statistic: 211.1 on 39 and 11636 DF,  p-value: < 2.2e-16


> # =========== Unumbered 3 Regressions in Appendix C.1 =====================
> # The first one
> lm_phy <- lm(log(cod_e) ~ log(product) + province
 .... [TRUNCATED]

> summary(lm_phy)

Call:
lm(formula = log(cod_e) ~ log(product) + province + industry_a +
    type_a, data = POL5[dm1_code_a == 1 & intensity > 0])

Residuals:
     Min       1Q   Median       3Q      Max
-13.7156  -1.3293   0.1174   1.4785   6.9085

Coefficients:
              Estimate Std. Error t value Pr(>|t|)
(Intercept)  -3.500637   0.403139  -8.683  < 2e-16 ***
log(product)  0.582348   0.013274  43.872  < 2e-16 ***
province12    0.846645   0.507475   1.668 0.095292 .
province13    3.096634   0.408158   7.587 3.71e-14 ***
province14    2.484245   0.441634   5.625 1.93e-08 ***
province15    2.871060   0.432071   6.645 3.27e-11 ***
province21    1.851277   0.429662   4.309 1.67e-05 ***
province22    1.844069   0.464755   3.968 7.33e-05 ***
province23    1.801350   0.457662   3.936 8.37e-05 ***
province31    0.367272   0.444393   0.826 0.408573
province32    1.174934   0.404203   2.907 0.003663 **
province33    1.803062   0.405948   4.442 9.07e-06 ***
province34    1.751814   0.460865   3.801 0.000145 ***
province35    1.828152   0.417334   4.381 1.20e-05 ***
province36    2.008088   0.423044   4.747 2.11e-06 ***
province37    1.402372   0.414149   3.386 0.000713 ***
province41    2.112781   0.444800   4.750 2.08e-06 ***
province42    2.493308   0.430359   5.794 7.19e-09 ***
province43    2.543748   0.406858   6.252 4.29e-10 ***
province44    1.232054   0.399293   3.086 0.002040 **
```

```
province45    3.088893   0.403751   7.650 2.27e-14 ***
province46    2.852169   0.559446   5.098 3.52e-07 ***
province50    0.981104   0.425045   2.308 0.021015 *
province51    2.016241   0.405682   4.970 6.86e-07 ***
province52    2.532615   0.453468   5.585 2.43e-08 ***
province53    2.635054   0.413783   6.368 2.04e-10 ***
province61    2.094637   0.443755   4.720 2.40e-06 ***
province62    3.811586   0.432909   8.805  < 2e-16 ***
province63    3.401091   0.637907   5.332 1.00e-07 ***
province64    4.139412   0.459758   9.003  < 2e-16 ***
province65    2.278728   0.423618   5.379 7.73e-08 ***
industry_a15 -0.111130   0.162809  -0.683 0.494896
industry_a17 -0.293487   0.112760  -2.603 0.009267 **
industry_a22  0.562485   0.078578   7.158 9.01e-13 ***
industry_a26 -2.453962   0.072406 -33.892  < 2e-16 ***
type_a3      -0.136418   0.060497  -2.255 0.024166 *
type_a4      -0.553279   0.174231  -3.176 0.001502 **
type_a5      -0.006735   0.133354  -0.051 0.959723
---
Signif. codes:  0 ?**?0.001 ?*?0.01 ??0.05 ??0.1 ??1

Residual standard error: 2.249 on 6900 degrees of freedom
Multiple R-squared:  0.3888,    Adjusted R-squared:  0.3855
F-statistic: 118.6 on 37 and 6900 DF,  p-value: < 2.2e-16


> # The second one
> lm_chem <- lm(log(cod_e) ~ log(product) + province
+               + industry_a + type_a,
+               data = POL5[dm1_code_a  .... [TRUNCATED]

> summary(lm_chem)

Call:
lm(formula = log(cod_e) ~ log(product) + province + industry_a +
    type_a, data = POL5[dm1_code_a == 2 & intensity > 0])

Residuals:
    Min      1Q  Median      3Q     Max
-9.8953 -1.1604  0.2451  1.3255  7.3782

Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept)  -4.58718    0.49127  -9.337  < 2e-16 ***
log(product)  0.77137    0.01837  41.990  < 2e-16 ***
province12    0.57384    0.54098   1.061 0.288858
province13    1.11495    0.46934   2.376 0.017563 *
province14    2.11332    0.52997   3.988 6.78e-05 ***
province15    1.76706    0.65011   2.718 0.006590 **
province21    1.18859    0.47385   2.508 0.012163 *
```

```
province22    0.44509    0.55706   0.799 0.424334
province23    0.93514    0.56862   1.645 0.100123
province31   -0.35811    0.48449  -0.739 0.459862
province32    0.43475    0.45875   0.948 0.343340
province33    1.03523    0.45649   2.268 0.023388 *
province34    0.92300    0.48814   1.891 0.058702 .
province35    0.96699    0.47187   2.049 0.040492 *
province36    1.41880    0.48978   2.897 0.003787 **
province37    0.59127    0.46660   1.267 0.205154
province41    0.54242    0.48801   1.111 0.266415
province42    1.17812    0.49269   2.391 0.016834 *
province43    1.37782    0.47316   2.912 0.003609 **
province44   -0.08719    0.45725  -0.191 0.848777
province45    1.22942    0.47623   2.582 0.009865 **
province46    0.04499    1.14165   0.039 0.968570
province50    1.48362    0.51536   2.879 0.004010 **
province51    0.93969    0.47376   1.983 0.047373 *
province52    0.94591    0.60478   1.564 0.117873
province53    1.26253    0.50438   2.503 0.012343 *
province54   -2.50401    2.18486  -1.146 0.251823
province61    1.08927    0.57867   1.882 0.059847 .
province62    1.59209    0.73537   2.165 0.030436 *
province63    1.93016    1.04072   1.855 0.063712 .
province64    1.92502    0.71889   2.678 0.007437 **
province65    2.20191    0.63455   3.470 0.000525 ***
industry_a15  0.31078    0.41469   0.749 0.453641
industry_a17  0.67574    0.17430   3.877 0.000107 ***
industry_a22  1.67873    0.16256  10.327  < 2e-16 ***
industry_a26 -1.60910    0.16327  -9.855  < 2e-16 ***
type_a3      -0.20871    0.07240  -2.883 0.003962 **
type_a4      -0.74229    0.14168  -5.239 1.68e-07 ***
type_a5      -0.65906    0.13443  -4.903 9.77e-07 ***
---
Signif. codes:  0 ?**?0.001 ?*?0.01 ??0.05 ??0.1 ??1

Residual standard error: 2.098 on 4699 degrees of freedom
Multiple R-squared:  0.4799,    Adjusted R-squared:  0.4757
F-statistic: 114.1 on 38 and 4699 DF,  p-value: < 2.2e-16


> # ============ Fixed Cost/Output Ratio of Clean Firms =====================
> # Used in calibration
> sel <- which(POL5$dm1_code_a == 4 | POL5$dm1_ .... [TRUNCATED]

> sum(POL5$dm1_inv[sel], na.rm = TRUE)/sum(POL5$product[sel], na.rm = TRUE)
[1] 0.02500509

> # ==========================================================================
> #                   3. Firm Size Distribution
> # ================== .... [TRUNCATED]
```

```
> load("./Data/CNEC_avgp.RData")

> CHNall <- CNEC_avgp

> rm(CNEC_avgp)

> USall <- read.csv("./Data/susb04.csv",header = TRUE)

> USall <- as.data.table(USall)

> USall <- USall[,list(NAICS, ENTRSIZE, FIRM, ESTB,
+         EMPL, NAICSDSCR, ENTRSIZEDSCR)]

> qup <- quantile(CHNall$nbarworkers, probs=c(.995),na.rm= TRUE)

> qdown <- quantile(CHNall$nbarworkers, probs=c(.01),na.rm= TRUE)

> sel <- which(CHNall$nbarworkers > 0 & CHNall$nbarworkers < qup
+              & CHNall$nbarworkers > qdown)

> CHNall <- CHNall[sel]

> # ---------------------------------------------------------------------------
> #                  Pooled Polluting
> # ---------------------------- .... [TRUNCATED]

> CH <- CHNall[sel]

> sel <- which(USall$NAICS == 3221 | USall$NAICS == 311
+     | USall$NAICS == 313 | USall$NAICS == 3251
+     | USall$NAICS == 3252 | USall$NAICS = .... [TRUNCATED]

> US <- USall[sel]

> # Process the U.S. Data
> sel <- which(US$ENTRSIZE != 1 & US$ENTRSIZE != 6 &  US$ENTRSIZE != 9)

> US <- US[sel]

> USSUM <- US[, list(FIRM=sum(FIRM, na.rm = TRUE),
+         ESTB=sum(ESTB, na.rm = TRUE), EMPL=sum(EMPL, na.rm = TRUE)),
+         by=list(ENTRSIZE .... [TRUNCATED]

> cf <- c(0,4,9,14,19,24,29,34,39,44,49,74,99,149,199,299,
+         399,499,749,999,1499,2499,5000)

> USSUM <- within(USSUM,
+                 {AVGF <- EMPL/FIRM   # Average firm size
+                 AVGE <- EMPL/ESTB   # Average plant size
```

```
+       .... [TRUNCATED]

> # Calculate the employment share
> cutoff <- c(1,19,99,399)

> n1 <- length(cutoff)

> distchn <- rep(0,n1)

> distus <- distchn

> for (i in 2:n1){
+   sel <- which(USSUM$AVGF > cutoff[i-1] & USSUM$AVGF <= cutoff[i])
+   distus[i-1] <- sum(USSUM$EMPL[sel])
+   sel1 <- which(CH$n .... [TRUNCATED]

> # Last group
> sel <- which(USSUM$AVGF > cutoff[n1])

> distus[n1] <- sum(USSUM$EMPL[sel])

> distus <- distus/sum(distus)

> sel1 <- which(CH$nbarworkers > cutoff[n1])

> distchn[n1] <- sum(CH$nbarworkers[sel1])

> distchn <- distchn/sum(distchn)

> # ================== Figure 2 ===========================================
> pdf("./Results/Figure2_Left.pdf",height=6,width=7.5)

> barplot(rbind(distchn,distus),beside=TRUE,col=c("red","blue"),
+     ylim=c(0,1.0),xlab="Firm Size",main="Pooled Polluting",
+     cex.main = 2.50, .... [TRUNCATED]

> title(ylab = "Employment Share", line = 2.3, cex.lab = 1.5)

> legend("topleft", c("China","US"),fill=c("red","blue"),
+        bty="o",cex=1.5)

> dev.off()
null device
          1

> # Size Distribution by groups used in computation
> cutoff <- c(1,19,49,99,399)

> n1 <- length(cutoff)
```

```
> distchn <- rep(0,n1)

> distus <- distchn

> for (i in 2:n1){
+   sel <- which(USSUM$AVGF > cutoff[i-1] & USSUM$AVGF <= cutoff[i])
+   distus[i-1] <- sum(USSUM$EMPL[sel])
+   sel1 <- which(CH$n .... [TRUNCATED]

> # Last category
> sel <- which(USSUM$AVGF > cutoff[n1])

> distus[n1] <- sum(USSUM$EMPL[sel])

> distus <- distus/sum(distus)

> sel1 <- which(CH$nbarworkers > cutoff[n1])

> distchn[n1] <- sum(CH$nbarworkers[sel1])

> distchn <- distchn/sum(distchn)

> # Employment distribution for polluting industries
> distchn
[1] 0.0906827 0.1594206 0.1597259 0.3495389 0.2406319

> # --------------------------------------------------------------------------
> #                   All Manufacturing
> # -------------------------- .... [TRUNCATED]

> sel <- which(USall$NAICS == "31-33")

> US <- USall[sel]

> # Process the U.S. Data
> sel <- which(US$ENTRSIZE != 1 & US$ENTRSIZE != 6 &  US$ENTRSIZE != 9)

> US <- US[sel]

> USSUM <- US[, list(FIRM=sum(FIRM, na.rm = TRUE),
+     ESTB=sum(ESTB, na.rm = TRUE), EMPL=sum(EMPL, na.rm = TRUE)),
+     by=list(ENTRSIZE)]

> cf <- c(0,4,9,14,19,24,29,34,39,44,49,74,99,149,199,299,
+         399,499,749,999,1499,2499,5000)

> USSUM <- within(USSUM,
+                 {AVGF <- EMPL/FIRM   # Average firm size
+                 AVGE <- EMPL/ESTB   # Average plant size
+       .... [TRUNCATED]
```

```
> # Calculate the employment share
> cutoff <- c(1,19,99,399)

> n1 <- length(cutoff)

> distchn <- rep(0,n1)

> distus <- distchn

> for (i in 2:n1){
+   sel <- which(USSUM$AVGF > cutoff[i-1] & USSUM$AVGF <= cutoff[i])
+   distus[i-1] <- sum(USSUM$EMPL[sel])
+   sel1 <- which(CH$n .... [TRUNCATED]

> # Last category
> sel <- which(USSUM$AVGF > cutoff[n1])

> distus[n1] <- sum(USSUM$EMPL[sel])

> distus <- distus/sum(distus)

> sel1 <- which(CH$nbarworkers > cutoff[n1])

> distchn[n1] <- sum(CH$nbarworkers[sel1])

> distchn <- distchn/sum(distchn)

> # =================== Figure 2 ==========================================
> pdf("./Results/Figure2_Right.pdf",height=6,width=7.5)

> barplot(rbind(distchn,distus),beside=TRUE,col=c("red","blue"),
+     ylim=c(0,1.0),xlab="Firm Size",main="All Manufacturing",
+     cex.main = 2.50 .... [TRUNCATED]

> title(ylab = "Employment Share", line = 2.3, cex.lab = 1.5)

> legend("topleft", c("China","US"),fill=c("red","blue"),
+        bty="o",cex=1.5)

> dev.off()
null device
          1

> # -----------------------------------------------------------------
> # Compute the calibration target of employment distribution
> # -------------- .... [TRUNCATED]

> n1 <- length(cutoff)
```

```
> distchn <- rep(0,n1)

> distus <- distchn

> for (i in 2:n1){
+   sel <- which(USSUM$AVGF > cutoff[i-1] & USSUM$AVGF <= cutoff[i])
+   distus[i-1] <- sum(USSUM$EMPL[sel])
+   sel1 <- which(CH$n .... [TRUNCATED]

> # Last category
> sel <- which(USSUM$AVGF > cutoff[n1])

> distus[n1] <- sum(USSUM$EMPL[sel])

> distus <- distus/sum(distus)

> sel1 <- which(CH$nbarworkers > cutoff[n1])

> distchn[n1] <- sum(CH$nbarworkers[sel1])

> distchn <- distchn/sum(distchn)

> # =================== Calibration Target ===================================
> # Employment share used in quantitative part
> distchn
[1] 0.08640851 0.16107471 0.16553707 0.35010698 0.23687273

> # ------------------------------------------------------------------
> # Compute the calibration target of firm size distribution
> # ---------------- .... [TRUNCATED]

> n1 <- length(cutoff)

> distchn <- rep(0,n1)

> # distus <- distchn
>
> for (i in 2:n1){
+   sel1 <- which(CH$nbarworkers > cutoff[i-1]
+           & CH$nbarworkers <= cutoff[i])
+   distchn[i-1 .... [TRUNCATED]

> sel1 <- which(CH$nbarworkers > cutoff[n1])

> distchn[n1] <- length(sel1)

> distchn <- distchn/sum(distchn)

> # =================== Calibration Target ===================================
> # Firm size distribution used in quantitative part
```

```
> distchn
[1] 0.46980903 0.27976175 0.12895600 0.10196701 0.01950621

> # ==========================================================================
> #                       4. Distortions
> # ========================== .... [TRUNCATED]

> load("./Data/CNEC_avgp.RData")

> CNEC <- CNEC_avgp

> rm(CNEC_avgp)

> # Drop irregular samples
> CNEC <- CNEC[status == 1]

> CNEC <- CNEC[product > 0]

> CNEC <- CNEC[totcapital > 0]

> CNEC <- CNEC[nbarworkers > 0]

> CNEC <- CNEC[wage + nonwage > 0]

> # Construct categorical variables
> CNEC$type_a <- factor(CNEC$type_a)

> CNEC$province <- factor(CNEC$province)

> CNEC$industry <- factor(CNEC$industry)

> CNEC$industry_a <- factor(CNEC$industry_a)

> # Calculate new variables
> CNEC <- within(CNEC,
+       {lcomp <- wage + nonwage
+       age <- 2005 - founding_y
+       })

> # Calculate average factor products
> alpha = 0.5376

> gamma = 0.93

> CNEC <- within(CNEC,
+       {phik   <- product/totcapital
+        phil   <- product/lcomp
+        phil_1 <- product/nbarworkers
+        phi    < .... [TRUNCATED]
```

```
> philz <- CNEC

> # Five Polluting Industries
> sel <- which(philz$industry_a == 22 | philz$industry_a == 13
+       | philz$industry_a == 15 | philz$industry_a == 1 .... [TRUNCATED]

> philz_pol <- philz[sel]

> cutup = 0.90

> cutdown = 0.10

> phiup <- quantile(philz_pol$phil, probs=c(cutup), na.rm = TRUE)

> phidown <- quantile(philz_pol$phil, probs=c(cutdown), na.rm = TRUE)

> zup <- quantile(philz_pol$z, probs=c(cutup), na.rm = TRUE)

> zdown <- quantile(philz_pol$z, probs=c(cutdown), na.rm = TRUE)

> sel <- which(philz_pol$z > zup)

> zupnew <- mean(philz_pol$z[sel])

> sel <- which(philz_pol$z < zdown)

> zdownnew <- mean(philz_pol$z[sel])

> sel <- which(philz_pol$phil > phiup)

> phiupnew <- mean(philz_pol$phil[sel])

> sel <- which(philz_pol$phil < phidown)

> phidownnew <- mean(philz_pol$phil[sel])

> phi_quant <- (log(phidownnew/phiupnew))/(log(zupnew/zdownnew))

> # Five polluting industries only
> phi_quant
[1] -0.02887093

> # All Manufacturing Industries
> cutup = 0.90

> cutdown = 0.10

> phiup <- quantile(CNEC$phil, probs=c(cutup), na.rm = TRUE)

> phidown <- quantile(CNEC$phil, probs=c(cutdown), na.rm = TRUE)
```

```
> zup <- quantile(CNEC$z, probs=c(cutup), na.rm = TRUE)

> zdown <- quantile(CNEC$z, probs=c(cutdown), na.rm = TRUE)

> sel <- which(CNEC$z > zup)

> zupnew <- mean(CNEC$z[sel])

> sel <- which(CNEC$z < zdown)

> zdownnew <- mean(CNEC$z[sel])

> sel <- which(CNEC$phil > phiup)

> phiupnew <- mean(CNEC$phil[sel])

> sel <- which(CNEC$phil < phidown)

> phidownnew <- mean(CNEC$phil[sel])

> phi_quant <- (log(phidownnew/phiupnew))/(log(zupnew/zdownnew))

> # ================== phi1 in calibration ==================================
> # All manufacturing
> phi_quant
[1] -0.02811157

> # -------------------------------------------------------------------
> # Plot Figure 3
> # ---------------------------------------------------------- .... [TRUNCATED]

> load("./Data/CNEC_avgp.RData")

> CNEC <- CNEC_avgp

> rm(CNEC_avgp)

> # Drop irregular samples
> CNEC <- CNEC[status == 1]

> CNEC <- CNEC[product > 0]

> CNEC <- CNEC[totcapital > 0]

> CNEC <- CNEC[nbarworkers > 0]

> CNEC <- CNEC[wage + nonwage > 0]

> # Construct categorical variables
```

```
> CNEC$type_a <- factor(CNEC$type_a)

> CNEC$province <- factor(CNEC$province)

> CNEC$industry <- factor(CNEC$industry)

> CNEC$industry_a <- factor(CNEC$industry_a)

> # Calculate new variables
> CNEC <- within(CNEC,
+                {lcomp <- wage + nonwage
+                 age <- 2005 - founding_y
+                .... [TRUNCATED]

> # Calculate average factor products
> alpha = 0.5376

> gamma = 0.93

> CNEC <- within(CNEC,
+                {phik   <- product/totcapital
+                 phil   <- product/lcomp
+                 phil_1 <- product/nbar .... [TRUNCATED]

> CNEC_RSV = CNEC

> sel <- which(CNEC$industry_a == 22 | CNEC$industry_a == 13
+              | CNEC$industry_a == 15 | CNEC$industry_a == 17
+              | CNEC$in .... [TRUNCATED]

> CNEC <- CNEC[sel]

> CNEC_NPOL <- CNEC_RSV[-sel]

> # now CNEC is polluting industries
> qup <- quantile(CNEC$phi, probs=c(.975),na.rm=TRUE)

> qdown <- quantile(CNEC$phi, probs=c(0.025),na.rm=TRUE)

> sel <- which(CNEC$phi>qdown & CNEC$phi<qup)

> CNEC_TRIM <- CNEC[sel]

> qup <- quantile(CNEC_TRIM$z, probs=c(.975),na.rm=TRUE)

> qdown <- quantile(CNEC_TRIM$z, probs=c(0.025),na.rm=TRUE)

> sel <- which(CNEC_TRIM$z>qdown & CNEC_TRIM$z<qup)

> CNEC_TRIM <- CNEC_TRIM[sel]
```

```
> CNEC_TRIM <- within(CNEC_TRIM,
+                     {logzratio <- log(z)/(mean(log(z),na.rm = TRUE))
+                     logphiratio <- log(phi)/ .... [TRUNCATED]

> # Calculate quintiles
> cutoff <- seq(from = 0.01, to = 0.99, by = 0.04)

> qcut <- quantile(CNEC_TRIM$logzratio, probs = cutoff, na.rm = TRUE)

> n1 <- length(cutoff)

> zplot <- rep(0,n1-1)

> zplotraw <- zplot

> phiplot <- zplot

> phiplotraw <- zplot

> for (i in 2:n1){
+   sel <- which(CNEC_TRIM$logzratio > qcut[i-1]
+                & CNEC_TRIM$logzratio <= qcut[i])
+   zplot[i-1] <-
+     sum(C .... [TRUNCATED]

> # =================== Figure 3 ===========================================
> pdf("./Results/Figure3_Left.pdf",height=5,width=5)

> plot(phiplot~zplot,cex=0.5,mgp=c(1.75, 0.75, 0),
+      xlab="Log Productivity",ylab="Log AFP",
+      main="Average Factor Product: \n Polluting S ..." ... [TRUNCATED]

> dev.off()
null device
          1

> # now CNEC is non-polluting industries
> CNEC <- CNEC_NPOL

> qup <- quantile(CNEC$phi, probs=c(.975),na.rm=TRUE)

> qdown <- quantile(CNEC$phi, probs=c(0.025),na.rm=TRUE)

> sel <- which(CNEC$phi>qdown & CNEC$phi<qup)

> CNEC_TRIM <- CNEC[sel]

> qup <- quantile(CNEC_TRIM$z, probs=c(.975),na.rm=TRUE)
```

```
> qdown <- quantile(CNEC_TRIM$z, probs=c(0.025),na.rm=TRUE)

> sel <- which(CNEC_TRIM$z>qdown & CNEC_TRIM$z<qup)

> CNEC_TRIM <- CNEC_TRIM[sel]

> CNEC_TRIM <- within(CNEC_TRIM,
+                      {logzratio <- log(z)/(mean(log(z),na.rm = TRUE))
+                      logphiratio <- log(phi)/ .... [TRUNCATED]

> # Calculate quintiles
> cutoff <- seq(from = 0.01, to = 0.99, by = 0.04)

> qcut <- quantile(CNEC_TRIM$logzratio, probs = cutoff, na.rm = TRUE)

> n1 <- length(cutoff)

> zplot <- rep(0,n1-1)

> zplotraw <- zplot

> phiplot <- zplot

> phiplotraw <- zplot

> for (i in 2:n1){
+   sel <- which(CNEC_TRIM$logzratio > qcut[i-1]
+                & CNEC_TRIM$logzratio <= qcut[i])
+   zplot[i-1] <-
+     sum(C .... [TRUNCATED]

> # =================== Figure 3 ============================================
> pdf("./Results/Figure3_Right.pdf",height=5,width=5)

> plot(phiplot~zplot,cex=0.5,mgp=c(1.75, 0.75, 0),
+      xlab="Log Productivity",ylab="Log AFP",
+      main="Average Factor Product: \n Non-Polluti ..." ... [TRUNCATED]

> dev.off()
null device
          1
```