The Granular Origins of Aggregate Fluctuations: Empirical Work

Compute Granular Residual

Alex Chinco

October 12, 2010

This code computes the granular residuals used in Xavier Gabaix's 2010 Econometrica article, The Granular Origins of Aggregate Fluctuations.

The code is organized as follows. In section 1 below I estimate innovations to firm productivity growth as deviations from year and year by industry means. In section 2 I use an alternative approach and estimate innovations to firm productivity growth as residuals from 6 regressions of the firm level productivity growth rate on year and year by industry mean productivity growth rates as well as interactions. In section 3 I use these estimates of the innovations to firm level productivity growth to compute the granular residuals.

I use the `R` packages listed below.

```
> rm(list = ls())
> library(foreign)
> library(matlab)
> library(xtable)
> library(reshape)
> library(ggplot2)
> library(plyr)
> library(plm)
> library(graphics)
```

These are the directories I use.

```
> HOME_DIRECTORY <- "~/Dropbox/raWork/granularOrigins/Granular-FinalMaterials/empiricalResults/"
> FIGURES_DIRECTORY <- "~/Dropbox/raWork/granularOrigins/Granular-FinalMaterials/empiricalResults/figures/"
> DATA_DIRECTORY <- "~/Dropbox/raWork/granularOrigins/Granular-FinalMaterials/data/"
```

1 Compute Innovations to Firm Productivity Growth as Demeaned Values

In this section, I compute innovations to firm productivity growth as deviations from the year and year by industry mean productivity growth rates. First, I compute the productivty growth rates. Then, I compute the mean productivity growth rates at the year and year by industry levels. Finally, I compute the demeaned productivity growth rates and winsorize the estimates.

Compute Productivity Growth Rates

I take the `CompustatData` data frame and compute the productivity level as the firm's sales in millions of dollars divided by the number of employees in millions. I also compute the lagged productivity level using the lagged sales and number of employees series. Using both these estimates I compute the productivity growth as the log difference.

```
> load(file = paste(DATA_DIRECTORY, "CleanCompustatData_10Oct2010.Rdata", sep = ""))
> CompustatData$laggedLogSales <- log(CompustatData$laggedSales)
> CompustatData$productivity <- CompustatData$sales/CompustatData$numberOfEmployees
> CompustatData$laggedProductivity <- CompustatData$laggedSales/CompustatData$laggedNumberOfEmployees
> CompustatData$changeInLogProductivity <- log(CompustatData$productivity) - log(CompustatData$laggedProductivity)
```

Estimate Mean Productivity Growth

I compute 4 different productivity growth rate means: year and year by industry means for the top 100 and the top 1000 firms as sorted by lagged real sales volume. I then merge these data series back onto the original cleaned COMPUSTAT data. If there is only 1 firm in its industry in a year, then the industry level mean computations will throw an `NA`. For these observations I fill the mean in with the year mean rather than the year by industry mean.

In all of the text below, Q denotes the number of firms over which the productivity growth rate means are taken; whereas, K denotes the number of firms I will sum over to compute the granular residual.

```
> MeanChangeInLogProductivityQ1000GroupedByYear <- ddply(CompustatData[CompustatData$inTop1000Firms == 1, ], c("year"),
      function(X) mean(X$changeInLogProductivity, na.rm = TRUE))
> MeanChangeInLogProductivityQ100GroupedByYear <- ddply(CompustatData[CompustatData$inTop100Firms == 1, ], c("year"),
      function(X) mean(X$changeInLogProductivity, na.rm = TRUE))
> names(MeanChangeInLogProductivityQ1000GroupedByYear) <- c("year", "meanChangeInLogProductivityQ1000GroupedByYear")
> names(MeanChangeInLogProductivityQ100GroupedByYear) <- c("year", "meanChangeInLogProductivityQ100GroupedByYear")
> MeanChangeInLogProductivityQ1000GroupedByYearAndIndustry <- ddply(CompustatData[CompustatData$inTop1000Firms == 1,
      ], c("year", "industry"), function(X) mean(X$changeInLogProductivity, na.rm = TRUE))
```

```
> MeanChangeInLogProductivityQ1000GroupedByYearAndIndustry <- ddply(CompustatData[CompustatData$inTop100Firms == 1,
      ], c("year", "industry"), function(X) mean(X$changeInLogProductivity, na.rm = TRUE))
> names(MeanChangeInLogProductivityQ1000GroupedByYearAndIndustry) <- c("year", "industry", "meanChangeInLogProductivityQ1000GroupedByYearAndIndustry")
> names(MeanChangeInLogProductivityQ100GroupedByYearAndIndustry) <- c("year", "industry", "meanChangeInLogProductivityQ100GroupedByYearAndIndustry")
> CompustatData <- merge(CompustatData, MeanChangeInLogProductivityQ1000GroupedByYear, by = c("year"), all.x = TRUE)
> CompustatData <- merge(CompustatData, MeanChangeInLogProductivityQ100GroupedByYear, by = c("year"), all.x = TRUE)
> CompustatData <- merge(CompustatData, MeanChangeInLogProductivityQ1000GroupedByYearAndIndustry, by = c("year", "industry"),
      all.x = TRUE)
> CompustatData <- merge(CompustatData, MeanChangeInLogProductivityQ100GroupedByYearAndIndustry, by = c("year", "industry"),
      all.x = TRUE)
> CompustatData[is.na(CompustatData$meanChangeInLogProductivityQ1000GroupedByYearAndIndustry), ]$meanChangeInLogProductivityQ1000GroupedByYearAndIndustry <- CompustatData[is.na(CompustatData$meanChangeInLogProduct
      ]$meanChangeInLogProductivityQ1000GroupedByYear
> CompustatData[is.na(CompustatData$meanChangeInLogProductivityQ100GroupedByYearAndIndustry), ]$meanChangeInLogProductivityQ100GroupedByYearAndIndustry <- CompustatData[is.na(CompustatData$meanChangeInLogProductiv
      ]$meanChangeInLogProductivityQ100GroupedByYear
```

Demeaned Productivity Growth

Next, I use the 4 estimates of the mean productivity growth rate at the year and year by industry level to demean the productivity growth rate series.

```
> CompustatData$unwinsorizedDemeanedChangeInLogProductivityUsingMeanQ1000GroupedByYear <- with(CompustatData, changeInLogProductivity -
      meanChangeInLogProductivityQ1000GroupedByYear)
> CompustatData$unwinsorizedDemeanedChangeInLogProductivityUsingMeanQ100GroupedByYear <- with(CompustatData, changeInLogProductivity -
      meanChangeInLogProductivityQ100GroupedByYear)
> CompustatData$unwinsorizedDemeanedChangeInLogProductivityUsingMeanQ1000GroupedByYearAndIndustry <- with(CompustatData,
      changeInLogProductivity - meanChangeInLogProductivityQ1000GroupedByYearAndIndustry)
> CompustatData$unwinsorizedDemeanedChangeInLogProductivityUsingMeanQ100GroupedByYearAndIndustry <- with(CompustatData,
      changeInLogProductivity - meanChangeInLogProductivityQ100GroupedByYearAndIndustry)
```

I then winsorize the demeaned productivity growth rates at the $\pm 20\%$ level.

```
> CompustatData$demeanedChangeInLogProductivityUsingMeanQ1000GroupedByYear <- CompustatData$unwinsorizedDemeanedChangeInLogProductivityUsingMeanQ1000GroupedByYear
> CompustatData$demeanedChangeInLogProductivityUsingMeanQ100GroupedByYear <- CompustatData$unwinsorizedDemeanedChangeInLogProductivityUsingMeanQ100GroupedByYear
> CompustatData$demeanedChangeInLogProductivityUsingMeanQ1000GroupedByYearAndIndustry <- CompustatData$unwinsorizedDemeanedChangeInLogProductivityUsingMeanQ1000GroupedByYearAndIndustry
> CompustatData$demeanedChangeInLogProductivityUsingMeanQ100GroupedByYearAndIndustry <- CompustatData$unwinsorizedDemeanedChangeInLogProductivityUsingMeanQ100GroupedByYearAndIndustry
> CompustatData[CompustatData$demeanedChangeInLogProductivityUsingMeanQ1000GroupedByYear > 0.2, ]$demeanedChangeInLogProductivityUsingMeanQ1000GroupedByYear <- 0.2
> CompustatData[CompustatData$demeanedChangeInLogProductivityUsingMeanQ1000GroupedByYear < -0.2, ]$demeanedChangeInLogProductivityUsingMeanQ1000GroupedByYear <- -0.2
> CompustatData[CompustatData$demeanedChangeInLogProductivityUsingMeanQ100GroupedByYear > 0.2, ]$demeanedChangeInLogProductivityUsingMeanQ100GroupedByYear <- 0.2
> CompustatData[CompustatData$demeanedChangeInLogProductivityUsingMeanQ100GroupedByYear < -0.2, ]$demeanedChangeInLogProductivityUsingMeanQ100GroupedByYear <- -0.2
> CompustatData[CompustatData$demeanedChangeInLogProductivityUsingMeanQ1000GroupedByYearAndIndustry > 0.2, ]$demeanedChangeInLogProductivityUsingMeanQ1000GroupedByYearAndIndustry <- 0.2
> CompustatData[CompustatData$demeanedChangeInLogProductivityUsingMeanQ1000GroupedByYearAndIndustry < -0.2, ]$demeanedChangeInLogProductivityUsingMeanQ1000GroupedByYearAndIndustry <- -0.2
> CompustatData[CompustatData$demeanedChangeInLogProductivityUsingMeanQ100GroupedByYearAndIndustry > 0.2, ]$demeanedChangeInLogProductivityUsingMeanQ100GroupedByYearAndIndustry <- 0.2
> CompustatData[CompustatData$demeanedChangeInLogProductivityUsingMeanQ100GroupedByYearAndIndustry < -0.2, ]$demeanedChangeInLogProductivityUsingMeanQ100GroupedByYearAndIndustry <- -0.2
```

2 Estimate Innovations to Firm Productivity Growth as Residuals

Now, I take an alternative approach to estimating innovations to firm productivity growth. Instead of simply demeaning the productivity growth rate data, I run a regression of the firm level productivity growth rate data on different combinations of the mean productivity growth rate and sales volume shocks.

First, I compute demeaned lagged log sales volume at the year and year by industry level. Then, in order to help with the formatting of the regression tables, I create a dictionary of variable names and load a regression table formatting class. Next, I estimate innovations to firm productivity growth as the residuals from 2 sets of 6 regressions. Finally, I discharge all but the top 100 firms in each year since I only needed the full 1000 firm sample in order to compute the year and year by industry means.

Compute Demeaned Lagged Log Sales

I compute year and year by industry demeaned lagged log sales. First, I take the log of the `laggedSales` variable in the `CompustatData` data frame. Then, I take the mean at the year and year by industry levels. For industry years with only 1 firm, I take the same approach as above and fill in the mean with the year mean rather than the year by industry mean. Finally, compute demeaned lagged log sales.

```
> MeanLaggedLogSalesQ1000GroupedByYear <- ddply(CompustatData[CompustatData$inTop1000Firms == 1, ], c("year"), function(X) mean(X$laggedLogSales,
      na.rm = TRUE))
> MeanLaggedLogSalesQ100GroupedByYear <- ddply(CompustatData[CompustatData$inTop100Firms == 1, ], c("year"), function(X) mean(X$laggedLogSales,
      na.rm = TRUE))
> names(MeanLaggedLogSalesQ1000GroupedByYear) <- c("year", "meanLaggedLogSalesQ1000GroupedByYear")
> names(MeanLaggedLogSalesQ100GroupedByYear) <- c("year", "meanLaggedLogSalesQ100GroupedByYear")
> MeanLaggedLogSalesQ1000GroupedByYearAndIndustry <- ddply(CompustatData[CompustatData$inTop1000Firms == 1, ], c("year",
      "industry"), function(X) mean(X$laggedLogSales, na.rm = TRUE))
> MeanLaggedLogSalesQ100GroupedByYearAndIndustry <- ddply(CompustatData[CompustatData$inTop100Firms == 1, ], c("year",
      "industry"), function(X) mean(X$laggedLogSales, na.rm = TRUE))
> names(MeanLaggedLogSalesQ1000GroupedByYearAndIndustry) <- c("year", "industry", "meanLaggedLogSalesQ1000GroupedByYearAndIndustry")
> names(MeanLaggedLogSalesQ100GroupedByYearAndIndustry) <- c("year", "industry", "meanLaggedLogSalesQ100GroupedByYearAndIndustry")
> CompustatData <- merge(CompustatData, MeanLaggedLogSalesQ1000GroupedByYear, by = c("year"), all.x = TRUE)
> CompustatData <- merge(CompustatData, MeanLaggedLogSalesQ100GroupedByYear, by = c("year"), all.x = TRUE)
> CompustatData <- merge(CompustatData, MeanLaggedLogSalesQ1000GroupedByYearAndIndustry, by = c("year", "industry"),
      all.x = TRUE)
> CompustatData <- merge(CompustatData, MeanLaggedLogSalesQ100GroupedByYearAndIndustry, by = c("year", "industry"),
      all.x = TRUE)
> CompustatData[is.na(CompustatData$meanLaggedLogSalesQ1000GroupedByYearAndIndustry), ]$meanLaggedLogSalesQ1000GroupedByYearAndIndustry <- CompustatData[is.na(CompustatData$meanLaggedLogSalesQ1000GroupedByYearAndI
      ]$meanLaggedLogSalesQ1000GroupedByYear
> CompustatData[is.na(CompustatData$meanLaggedLogSalesQ100GroupedByYearAndIndustry), ]$meanLaggedLogSalesQ100GroupedByYearAndIndustry <- CompustatData[is.na(CompustatData$meanLaggedLogSalesQ100GroupedByYearAndIndu
      ]$meanLaggedLogSalesQ100GroupedByYear
> CompustatData$demeanedLaggedLogSalesUsingMeanQ1000GroupedByYear <- with(CompustatData, laggedLogSales - meanLaggedLogSalesQ1000GroupedByYear)
> CompustatData$demeanedLaggedLogSalesUsingMeanQ100GroupedByYear <- with(CompustatData, laggedLogSales - meanLaggedLogSalesQ100GroupedByYear)
> CompustatData$demeanedLaggedLogSalesUsingMeanQ1000GroupedByYearAndIndustry <- with(CompustatData, laggedLogSales -
      meanLaggedLogSalesQ1000GroupedByYearAndIndustry)
> CompustatData$demeanedLaggedLogSalesUsingMeanQ100GroupedByYearAndIndustry <- with(CompustatData, laggedLogSales -
      meanLaggedLogSalesQ100GroupedByYearAndIndustry)
```

Load Regression Table Class

Next, I create a dictionary of variable names used in the regression tables below and also load a regression table class in order to help with the display of the output. The whole point here is to make the output from R to LaTeX as clean as possible.

Below, I create a dictionary of regression terms below with both the variable name in the R data as well as a clean expression suitable for LaTeX display.

```
> DictionaryOfVariableNames <- list(c("(Intercept)", "changeInLogPerCapitaGdp", "solowResidual", "hamiltonOilShocks",
      "romerMonetaryShocks", "riskFreeRate", "termSpread", "changeInLogProductivity", "meanChangeInLogProductivityQ100GroupedByYear",
      "demeanedLaggedLogSalesUsingMeanQ100GroupedByYear", "I(demeanedLaggedLogSalesUsingMeanQ100GroupedByYear^2)",
      "meanChangeInLogProductivityQ100GroupedByYear:demeanedLaggedLogSalesUsingMeanQ100GroupedByYear", "meanChangeInLogProductivityQ100GroupedByYear:I(demeanedLaggedLogSalesUsingMeanQ100GroupedByYear^2)",
      "meanChangeInLogProductivityQ100GroupedByYearAndIndustry", "demeanedLaggedLogSalesUsingMeanQ100GroupedByYearAndIndustry",
      "I(demeanedLaggedLogSalesUsingMeanQ100GroupedByYearAndIndustry^2)", "meanChangeInLogProductivityQ100GroupedByYearAndIndustry:demeanedLaggedLogSalesUsingMeanQ100GroupedByYearAndIndustry",
      "meanChangeInLogProductivityQ100GroupedByYearAndIndustry:I(demeanedLaggedLogSalesUsingMeanQ100GroupedByYearAndIndustry^2)",
      "meanChangeInLogProductivityQ1000GroupedByYear", "demeanedLaggedLogSalesUsingMeanQ1000GroupedByYear", "I(demeanedLaggedLogSalesUsingMeanQ1000GroupedByYear^2)",
      "meanChangeInLogProductivityQ1000GroupedByYear:demeanedLaggedLogSalesUsingMeanQ1000GroupedByYear", "meanChangeInLogProductivityQ1000GroupedByYear:I(demeanedLaggedLogSalesUsingMeanQ1000GroupedByYear^2)",
      "meanChangeInLogProductivityQ1000GroupedByYearAndIndustry", "demeanedLaggedLogSalesUsingMeanQ1000GroupedByYearAndIndustry",
      "I(demeanedLaggedLogSalesUsingMeanQ1000GroupedByYearAndIndustry^2)", "meanChangeInLogProductivityQ1000GroupedByYearAndIndustry:demeanedLaggedLogSalesUsingMeanQ1000GroupedByYearAndIndustry",
      "meanChangeInLogProductivityQ1000GroupedByYearAndIndustry:I(demeanedLaggedLogSalesUsingMeanQ1000GroupedByYearAndIndustry^2)"),
      c("(Intercept)", "\\text{GDP Growth}_{TIME}", "\\text{Solow Residual}_{TIME}", "\\text{Oil}_{TIME}", "\\text{Monetary}_{TIME}",
          "r_{TIME}", "\\text{Term Spread}_{TIME}", "g_{i,TIME}", "\\bar{g}_{TIME}", "(\\ln S_{i,TIME-1} - \\overline{\\ln S_{TIME-1}})",
          "(\\ln S_{i,TIME-1} - \\overline{\\ln S_{TIME-1}})^2", "(\\ln S_{i,TIME-1} - \\overline{\\ln S_{TIME-1}}) \\cdot \\bar{g}_{TIME}",
          "(\\ln S_{i,TIME-1} - \\overline{\\ln S_{TIME-1}})^2 \\cdot \\bar{g}_{TIME}", "\\bar{g}_{I(i),TIME},", "(\\ln S_{i,TIME-1} - \\overline{\\ln S_{I(i),TIME-1}})",
          "(\\ln S_{i,TIME-1} - \\overline{\\ln S_{I(i),TIME-1}})^2", "(\\ln S_{i,TIME-1} - \\overline{\\ln S_{I(i),TIME-1}}) \\cdot \\bar{g}_{I(i),TIME}",
          "(\\ln S_{i,TIME-1} - \\overline{\\ln S_{I(i),TIME-1}})^2 \\cdot \\bar{g}_{I(i),TIME}", "\\bar{g}_{TIME}",
          "(\\ln S_{i,TIME-1} - \\overline{\\ln S_{TIME-1}})", "(\\ln S_{i,TIME-1} - \\overline{\\ln S_{TIME-1}})^2",
          "(\\ln S_{i,TIME-1} - \\overline{\\ln S_{TIME-1}}) \\cdot \\bar{g}_{TIME}", "(\\ln S_{i,TIME-1} - \\overline{\\ln S_{TIME-1}})^2 \\cdot \\bar{g}_{TIME}",
          "\\bar{g}_{I(i),TIME},", "(\\ln S_{i,TIME-1} - \\overline{\\ln S_{I(i),TIME-1}})", "(\\ln S_{i,TIME-1} - \\overline{\\ln S_{I(i),TIME-1}})^2",
          "(\\ln S_{i,TIME-1} - \\overline{\\ln S_{I(i),TIME-1}}) \\cdot \\bar{g}_{I(i),TIME}", "(\\ln S_{i,TIME-1} - \\overline{\\ln S_{I(i),TIME-1}})^2 \\cdot \\bar{g}_{I(i),TIME}"))
```

I want to estimate a series of regressions, access the outputs and then print the results to the screen later on. The best way for me to do this is to create a separate `RegressionObject` class. I load in this class below. This class eats a list of regression formulas and a data frame and returns the regression results as well as a nicely formatted table.

```
> Stangle(paste(HOME_DIRECTORY, "createRegressionTable.Rnw", sep = ""))
> source("createRegressionTable.R")
```

Estimate Innovations to Productivity Growth Rates

Now, I estimate the innovations to firm productivity growth using 2 sets of 6 regressions. I treat the regression using only the top 100 firms first. I define the table caption and initialize the regression output object. The output object eats a caption, a dictionary of variable names as well as a list of formulas. The result is a regression object that houses all of the estimates in R as well as in an easily accesible `xtable` object.

```
> CAPTION <- "\\textit{\\textbf{Innovations to Productivity Growth: Q=100}: This table estimates the productivity growth rate of the top 100 firms in the COMPUSTAT database as sorted by lagged sales from 1952 to 2
> ProductivityGrowthRateInnovationsAsResidualsQ100 <- new("RegressionObject", data = CompustatData[CompustatData$inTop100Firms == 1, ],
      variableNameDictionary = DictionaryOfVariableNames, formulas = list(list(c("changeInLogProductivity"), c("meanChangeInLogProductivityQ100GroupedByYear")),
          list(c("changeInLogProductivity"), c("meanChangeInLogProductivityQ100GroupedByYearAndIndustry")), list(c("changeInLogProductivity"),
              c("meanChangeInLogProductivityQ100GroupedByYear", "demeanedLaggedLogSalesUsingMeanQ100GroupedByYear", "meanChangeInLogProductivityQ100GroupedByYear:demeanedLaggedLogSalesUsingMeanQ100GroupedByYear"))
          list(c("changeInLogProductivity"), c("meanChangeInLogProductivityQ100GroupedByYear", "demeanedLaggedLogSalesUsingMeanQ100GroupedByYear",
              "I(demeanedLaggedLogSalesUsingMeanQ100GroupedByYear^2)", "meanChangeInLogProductivityQ100GroupedByYear:demeanedLaggedLogSalesUsingMeanQ100GroupedByYear",
              "meanChangeInLogProductivityQ100GroupedByYear:I(demeanedLaggedLogSalesUsingMeanQ100GroupedByYear^2)")), list(c("changeInLogProductivity"),
              c("meanChangeInLogProductivityQ100GroupedByYear", "meanChangeInLogProductivityQ100GroupedByYearAndIndustry", "demeanedLaggedLogSalesUsingMeanQ100GroupedByYear",
                  "demeanedLaggedLogSalesUsingMeanQ100GroupedByYearAndIndustry", "meanChangeInLogProductivityQ100GroupedByYear:demeanedLaggedLogSalesUsingMeanQ100GroupedByYear",
                  "meanChangeInLogProductivityQ100GroupedByYearAndIndustry:demeanedLaggedLogSalesUsingMeanQ100GroupedByYearAndIndustry")),
          list(c("changeInLogProductivity"), c("meanChangeInLogProductivityQ100GroupedByYear", "meanChangeInLogProductivityQ100GroupedByYearAndIndustry",
              "demeanedLaggedLogSalesUsingMeanQ100GroupedByYear", "I(demeanedLaggedLogSalesUsingMeanQ100GroupedByYear^2)", "demeanedLaggedLogSalesUsingMeanQ100GroupedByYearAndIndustry",
              "I(demeanedLaggedLogSalesUsingMeanQ100GroupedByYearAndIndustry^2)", "meanChangeInLogProductivityQ100GroupedByYear:demeanedLaggedLogSalesUsingMeanQ100GroupedByYear",
              "meanChangeInLogProductivityQ100GroupedByYear:I(demeanedLaggedLogSalesUsingMeanQ100GroupedByYear^2)", "meanChangeInLogProductivityQ100GroupedByYearAndIndustry", "meanChangeInLogProductivityQ100GroupedByYear:I(demeanedLaggedLogSalesUsingMeanQ
              "meanChangeInLogProductivityQ100GroupedByYearAndIndustry:I(demeanedLaggedLogSalesUsingMeanQ100GroupedByYearAndIndustry^2)"))),
      lags = list(c(0)), caption = CAPTION)
```

Then, I write the resulting regression output to a LaTeXtable which I include below. The regression table object holds all of the results in an `xtable` object so all I have to do is access this class attribute.

```
> write(ProductivityGrowthRateInnovationsAsResidualsQ100@xtable, file = paste(FIGURES_DIRECTORY, "ResidualEstimatesOfInnovationsToFirmProductivityGrowthQ100_10Oct2010.tex",
      sep = ""))
```

Finally, I add the residuals to the existing dataset as a an additional 6 different estimates of the innovations to firm level productivity growth rates.

```
> for (REG in 1:ProductivityGrowthRateInnovationsAsResidualsQ100@numberOfFormulas) {
      NAME <- paste("residualChangeInLogProductivityQ100Equation", REG, sep = "")
      CompustatData[CompustatData$inTop100Firms == 1, NAME] <- ProductivityGrowthRateInnovationsAsResidualsQ100@results[[2]][[REG]]$residuals
      CompustatData[CompustatData$inTop100Firms == 1 & CompustatData[, NAME] > 0.2, NAME] <- 0.2
      CompustatData[CompustatData$inTop100Firms == 1 & CompustatData[, NAME] < -0.2, NAME] <- -0.2
  }
```

I repeat these 3 steps for the top 1000 firm subset as well. This gives me an additional 12 estimates of the innovations to firm level productivity growth.

```
> CAPTION <- "\\textit{\\textbf{Innovations to Productivity Growth: Q=1000}: This table estimates the productivity growth rate of the top 1000 firms in the COMPUSTAT database as sorted by lagged sales from 1952 to
> ProductivityGrowthRateInnovationsAsResidualsQ1000 <- new("RegressionObject", data = CompustatData[CompustatData$inTop1000Firms == 1,
      ], variableNameDictionary = DictionaryOfVariableNames, formulas = list(list(c("changeInLogProductivity"), c("meanChangeInLogProductivityQ1000GroupedByYear")),
      list(c("changeInLogProductivity"), c("meanChangeInLogProductivityQ1000GroupedByYearAndIndustry")), list(c("changeInLogProductivity"),
          c("meanChangeInLogProductivityQ1000GroupedByYear", "demeanedLaggedLogSalesUsingMeanQ1000GroupedByYear", "meanChangeInLogProductivityQ1000GroupedByYear:demeanedLaggedLogSalesUsingMeanQ1000GroupedByYear"))
      list(c("changeInLogProductivity"), c("meanChangeInLogProductivityQ1000GroupedByYear", "demeanedLaggedLogSalesUsingMeanQ1000GroupedByYear",
          "I(demeanedLaggedLogSalesUsingMeanQ1000GroupedByYear^2)", "meanChangeInLogProductivityQ1000GroupedByYear:demeanedLaggedLogSalesUsingMeanQ1000GroupedByYear",
          "meanChangeInLogProductivityQ1000GroupedByYear:I(demeanedLaggedLogSalesUsingMeanQ1000GroupedByYear^2)")), list(c("changeInLogProductivity"),
          c("meanChangeInLogProductivityQ1000GroupedByYear", "meanChangeInLogProductivityQ1000GroupedByYearAndIndustry", "demeanedLaggedLogSalesUsingMeanQ1000GroupedByYear",
              "demeanedLaggedLogSalesUsingMeanQ1000GroupedByYearAndIndustry", "meanChangeInLogProductivityQ1000GroupedByYear:demeanedLaggedLogSalesUsingMeanQ1000GroupedByYear",
              "meanChangeInLogProductivityQ1000GroupedByYearAndIndustry:demeanedLaggedLogSalesUsingMeanQ1000GroupedByYearAndIndustry")),
      list(c("changeInLogProductivity"), c("meanChangeInLogProductivityQ1000GroupedByYear", "meanChangeInLogProductivityQ1000GroupedByYearAndIndustry",
          "demeanedLaggedLogSalesUsingMeanQ1000GroupedByYear", "I(demeanedLaggedLogSalesUsingMeanQ1000GroupedByYear^2)", "demeanedLaggedLogSalesUsingMeanQ1000GroupedByYearAndIndustry",
          "I(demeanedLaggedLogSalesUsingMeanQ1000GroupedByYearAndIndustry^2)", "meanChangeInLogProductivityQ1000GroupedByYear:demeanedLaggedLogSalesUsingMeanQ1000GroupedByYear",
          "meanChangeInLogProductivityQ1000GroupedByYearAndIndustry:demeanedLaggedLogSalesUsingMeanQ1000GroupedByYearAndIndustry", "meanChangeInLogProductivityQ1000GroupedByYear:I(demeanedLaggedLogSalesUsingMeanQ1
```

```
        "meanChangeInLogProductivityQ1000GroupedByYearAndIndustry:I(demeanedLaggedLogSalesUsingMeanQ1000GroupedByYearAndIndustry^2)"))),
      lags = list(c(0)), caption = CAPTION)
> write(ProductivityGrowthRateInnovationsAsResidualsQ1000@xtable, file = paste(FIGURES_DIRECTORY, "ResidualEstimatesOfInnovationsToFirmProductivityGrowthQ1000_10Oct2010.tex",
      sep = ""))
> for (REG in 1:ProductivityGrowthRateInnovationsAsResidualsQ1000@numberOfFormulas) {
      NAME <- paste("residualChangeInLogProductivityQ1000Equation", REG, sep = "")
      CompustatData[CompustatData$inTop1000Firms == 1, NAME] <- ProductivityGrowthRateInnovationsAsResidualsQ1000@results[[2]][[REG]]$residuals
      CompustatData[CompustatData$inTop1000Firms == 1 & CompustatData[, NAME] > 0.2, NAME] <- 0.2
      CompustatData[CompustatData$inTop1000Firms == 1 & CompustatData[, NAME] < -0.2, NAME] <- -0.2
  }
```

	(1)	(2)	(3)	(4)	(5)	(6)
(Intercept)	7.2e-16	-7.9e-17	7.2e-16	0.0014	7.2e-18	0.0019
	(0.002)	(0.0013)	(0.002)	(0.0024)	(0.0017)	(0.0021)
$\bar{g}_t$	1**		1**	0.9**	-0.00097	-0.083
	(0.076)		(0.076)	(0.093)	(0.068)	(0.081)
$\bar{g}_{I(i),t}$,		1**			1**	0.97**
		(0.02)			(0.021)	(0.022)
$(\ln S_{i,t-1} - \overline{\ln S_{t-1}})$			0.0041	0.0076	-0.0031	0.00012
			(0.0032)	(0.0048)	(0.0043)	(0.0051)
$(\ln S_{i,t-1} - \overline{\ln S_{t-1}}) \cdot \bar{g}_t$			-0.15	-0.42*	0.17	-0.041
			(0.12)	(0.18)	(0.11)	(0.16)
$(\ln S_{i,t-1} - \overline{\ln S_{t-1}})^2$				-0.0034		-0.0012
				(0.0035)		(0.0042)
$(\ln S_{i,t-1} - \overline{\ln S_{t-1}})^2 \cdot \bar{g}_t$				0.26*		0.17
				(0.13)		(0.12)
$(\ln S_{i,t-1} - \overline{\ln S_{I(i),t-1}})$					0.0068	0.0089
					(0.0046)	(0.0047)
$(\ln S_{i,t-1} - \overline{\ln S_{I(i),t-1}}) \cdot \bar{g}_{I(i),t}$					-0.31**	-0.4**
					(0.05)	(0.057)
$(\ln S_{i,t-1} - \overline{\ln S_{I(i),t-1}})^2$						-0.0053
						(0.0045)
$(\ln S_{i,t-1} - \overline{\ln S_{I(i),t-1}})^2 \cdot \bar{g}_{I(i),t}$						0.16**
						(0.05)
N	5700	5700	5700	5700	5700	5700
R^2	0.0292	0.305	0.0295	0.0302	0.31	0.312
Adj. R^2	0.029	0.305	0.029	0.0293	0.309	0.311

Table 1: ***Innovations to Productivity Growth: Q=100****: This table estimates the productivity growth rate of the top 100 firms in the COMPUSTAT database as sorted by lagged sales from 1952 to 2008 at a yearly frquency using regressions on mean growth rates and interactions with firm size.*

	(1)	(2)	(3)	(4)	(5)	(6)
(Intercept)	3.2e-15	-1.3e-16	3.2e-15	-0.0021	2e-16	-0.0016
	(0.001)	(0.00079)	(0.001)	(0.0012)	(0.00096)	(0.0012)
$\bar{g}_t$	1**		1**	1**	-0.00053	0.0097
	(0.037)		(0.037)	(0.045)	(0.038)	(0.045)
$\bar{g}_{I(i),t}$,		1**			1**	1**
		(0.012)			(0.013)	(0.015)
$(\ln S_{i,t-1} - \overline{\ln S_{t-1}})$			0.00019	-0.0011	-0.0026	-0.0037
			(0.00091)	(0.001)	(0.002)	(0.002)
$(\ln S_{i,t-1} - \overline{\ln S_{t-1}}) \cdot \bar{g}_t$			-0.051	-0.043	0.15**	0.16**
			(0.033)	(0.037)	(0.034)	(0.037)
$(\ln S_{i,t-1} - \overline{\ln S_{t-1}})^2$				0.0017**		0.00071
				(0.00057)		(0.00093)
$(\ln S_{i,t-1} - \overline{\ln S_{t-1}})^2 \cdot \bar{g}_t$				-0.0098		-0.0085
				(0.02)		(0.021)
$(\ln S_{i,t-1} - \overline{\ln S_{I(i),t-1}})$					0.0033	0.0034
					(0.002)	(0.002)
$(\ln S_{i,t-1} - \overline{\ln S_{I(i),t-1}}) \cdot \bar{g}_{I(i),t}$					-0.22**	-0.22**
					(0.014)	(0.015)
$(\ln S_{i,t-1} - \overline{\ln S_{I(i),t-1}})^2$						7e-04
						(0.00099)
$(\ln S_{i,t-1} - \overline{\ln S_{I(i),t-1}})^2 \cdot \bar{g}_{I(i),t}$						-2.4e-05
						(0.0093)
N	52895	52895	52895	52895	52895	52895
R^2	0.0134	0.113	0.0135	0.0137	0.117	0.117
Adj. R^2	0.0134	0.113	0.0134	0.0136	0.117	0.117

Table 2: ***Innovations to Productivity Growth: Q=1000****: This table estimates the productivity growth rate of the top 1000 firms in the COMPUSTAT database as sorted by lagged sales from 1952 to 2008 at a yearly frquency using regressions on mean growth rates and interactions with firm size. There are not* 57000 *observations because in the years prior to* 1960 *there are fewer than* 1000 *firms in the cleaned COMPUSTAT data.*

Keep Only the Top 100 Firms in Each Year

Finally, I keep only the top 100 firms in each year as sorted by lagged sales volume. I only needed the full 1000 firm sample in order to compute the year and year by industry means. There is no further reason for me to haul around the extra 900 firms a year.

```
> CompustatData <- CompustatData[CompustatData$inTop100Firms == 1, ]
```

3 Compute Granular Residuals

Finally, I compute the granular residuals. First, I compute each firm's individual contribution to the granular residual. Then, I sum over all firms in each year to compute the full granular residual. I conclude the section by plotting the granular residual estimates.

Compute Firm Contribution to Granular Residuals

For each of the top 100 firms as sorted by lagged sales, I take the year and year by industry mean productivity growth rates and I compute the weighted sum of the demeaned productivity growth rates weighted by lagged sales and divide this quantity by the lagged real GDP. Each component of this sum represents a firm's contribution to the granular residual.

$$\gamma_t^i = \frac{S_{i,t-1}}{Y_{t-1}} \left(g_{i,t} - \bar{g}_t\right)$$
$$\gamma_t^i = \frac{S_{i,t-1}}{Y_{t-1}} \left(g_{i,t} - \bar{g}_{I(i),t}\right)$$

```
> load(file = paste(DATA_DIRECTORY, "CleanMacroeconomicData_10Oct2010.Rdata", sep = ""))
> LaggedRealGdp <- as.data.frame(cbind(MacroeconomicData$year, c(NA, MacroeconomicData[MacroeconomicData$year != 2008, ]$realGdp)))
> names(LaggedRealGdp) <- c("year", "laggedRealGdp")
> CompustatData <- merge(CompustatData, LaggedRealGdp, by = "year", all.x = TRUE)
> CompustatData$firmContributionToGranularResidualUsingYearDemeaningQ100 <- with(CompustatData, laggedSales * demeanedChangeInLogProductivityUsingMeanQ100GroupedByYear/laggedRealGdp)
> CompustatData$firmContributionToGranularResidualUsingYearAndIndustryDemeaningQ100 <- with(CompustatData, laggedSales * demeanedChangeInLogProductivityUsingMeanQ100GroupedByYearAndIndustry/laggedRealGdp)
> CompustatData$firmContributionToGranularResidualUsingResidualEquation1Q100 <- with(CompustatData, laggedSales * residualChangeInLogProductivityQ100Equation1/laggedRealGdp)
> CompustatData$firmContributionToGranularResidualUsingResidualEquation2Q100 <- with(CompustatData, laggedSales * residualChangeInLogProductivityQ100Equation2/laggedRealGdp)
> CompustatData$firmContributionToGranularResidualUsingResidualEquation3Q100 <- with(CompustatData, laggedSales * residualChangeInLogProductivityQ100Equation3/laggedRealGdp)
> CompustatData$firmContributionToGranularResidualUsingResidualEquation4Q100 <- with(CompustatData, laggedSales * residualChangeInLogProductivityQ100Equation4/laggedRealGdp)
> CompustatData$firmContributionToGranularResidualUsingResidualEquation5Q100 <- with(CompustatData, laggedSales * residualChangeInLogProductivityQ100Equation5/laggedRealGdp)
> CompustatData$firmContributionToGranularResidualUsingResidualEquation6Q100 <- with(CompustatData, laggedSales * residualChangeInLogProductivityQ100Equation6/laggedRealGdp)
> CompustatData$firmContributionToGranularResidualUsingYearDemeaningQ1000 <- with(CompustatData, laggedSales * demeanedChangeInLogProductivityUsingMeanQ1000GroupedByYear/laggedRealGdp)
> CompustatData$firmContributionToGranularResidualUsingYearAndIndustryDemeaningQ1000 <- with(CompustatData, laggedSales * demeanedChangeInLogProductivityUsingMeanQ1000GroupedByYearAndIndustry/laggedRealGdp)
> CompustatData$firmContributionToGranularResidualUsingResidualEquation1Q1000 <- with(CompustatData, laggedSales * residualChangeInLogProductivityQ1000Equation1/laggedRealGdp)
> CompustatData$firmContributionToGranularResidualUsingResidualEquation2Q1000 <- with(CompustatData, laggedSales * residualChangeInLogProductivityQ1000Equation2/laggedRealGdp)
> CompustatData$firmContributionToGranularResidualUsingResidualEquation3Q1000 <- with(CompustatData, laggedSales * residualChangeInLogProductivityQ1000Equation3/laggedRealGdp)
> CompustatData$firmContributionToGranularResidualUsingResidualEquation4Q1000 <- with(CompustatData, laggedSales * residualChangeInLogProductivityQ1000Equation4/laggedRealGdp)
> CompustatData$firmContributionToGranularResidualUsingResidualEquation5Q1000 <- with(CompustatData, laggedSales * residualChangeInLogProductivityQ1000Equation5/laggedRealGdp)
> CompustatData$firmContributionToGranularResidualUsingResidualEquation6Q1000 <- with(CompustatData, laggedSales * residualChangeInLogProductivityQ1000Equation6/laggedRealGdp)
```

Compute Granular Residuals

Next, I sum over all of the firm level contributions in each year to get the full granular residual estimate.

$$\Gamma_t^K = \sum_{i=1}^{K} \gamma_t^i$$

$$\Gamma_t^K = \sum_{i=1}^{K} \frac{S_{i,t-1}}{Y_{t-1}} \left(g_{i,t} - \bar{g}_t\right)$$

$$\Gamma_t^K = \sum_{i=1}^{K} \frac{S_{i,t-1}}{Y_{t-1}} \left(g_{i,t} - \bar{g}_{I(i),t}\right)$$

First, I compute the granular residual estimates that use $K = 100$.

```
> GranularResidualsK100 <- CompustatData[, c("year", "firmContributionToGranularResidualUsingYearDemeaningQ100", "firmContributionToGranularResidualUsingYearAndIndustryDemeaningQ100",
      "firmContributionToGranularResidualUsingResidualEquation1Q100", "firmContributionToGranularResidualUsingResidualEquation2Q100",
      "firmContributionToGranularResidualUsingResidualEquation3Q100", "firmContributionToGranularResidualUsingResidualEquation4Q100",
      "firmContributionToGranularResidualUsingResidualEquation5Q100", "firmContributionToGranularResidualUsingResidualEquation6Q100",
      "firmContributionToGranularResidualUsingYearDemeaningQ1000", "firmContributionToGranularResidualUsingYearAndIndustryDemeaningQ1000",
      "firmContributionToGranularResidualUsingResidualEquation1Q1000", "firmContributionToGranularResidualUsingResidualEquation2Q1000",
      "firmContributionToGranularResidualUsingResidualEquation3Q1000", "firmContributionToGranularResidualUsingResidualEquation4Q1000",
      "firmContributionToGranularResidualUsingResidualEquation5Q1000", "firmContributionToGranularResidualUsingResidualEquation6Q1000")]
> GranularResidualsK100 <- ddply(GranularResidualsK100, c("year"), function(X) colSums(X[, -1], na.rm = TRUE))
> names(GranularResidualsK100) <- c("year", "GranularResidualUsingYearDemeaningQ100K100", "GranularResidualUsingYearAndIndustryDemeaningQ100K100",
      "GranularResidualUsingResidualEquation1Q100K100", "GranularResidualUsingResidualEquation2Q100K100", "GranularResidualUsingResidualEquation3Q100K100",
      "GranularResidualUsingResidualEquation4Q100K100", "GranularResidualUsingResidualEquation5Q100K100", "GranularResidualUsingResidualEquation6Q100K100",
      "GranularResidualUsingYearDemeaningQ1000K100", "GranularResidualUsingYearAndIndustryDemeaningQ1000K100", "GranularResidualUsingResidualEquation1Q1000K100",
      "GranularResidualUsingResidualEquation2Q1000K100", "GranularResidualUsingResidualEquation3Q1000K100", "GranularResidualUsingResidualEquation4Q1000K100",
      "GranularResidualUsingResidualEquation5Q1000K100", "GranularResidualUsingResidualEquation6Q1000K100")
```

Then, I compute the granular residual estimates that use $K = 5$. For these estimates, I sum over just the top 5 firms as sorted by lagged sales volume rather than the full 100 firms each year.

```
> GranularResidualsK5 <- CompustatData[CompustatData$inTop5Firms == 1, c("year", "firmContributionToGranularResidualUsingYearDemeaningQ100",
      "firmContributionToGranularResidualUsingYearAndIndustryDemeaningQ100", "firmContributionToGranularResidualUsingResidualEquation1Q100",
      "firmContributionToGranularResidualUsingResidualEquation2Q100", "firmContributionToGranularResidualUsingResidualEquation3Q100",
      "firmContributionToGranularResidualUsingResidualEquation4Q100", "firmContributionToGranularResidualUsingResidualEquation5Q100",
      "firmContributionToGranularResidualUsingResidualEquation6Q100")]
> GranularResidualsK5 <- ddply(GranularResidualsK5, c("year"), function(X) colSums(X[, -1], na.rm = TRUE))
> names(GranularResidualsK5) <- c("year", "GranularResidualUsingYearDemeaningQ100K5", "GranularResidualUsingYearAndIndustryDemeaningQ100K5",
      "GranularResidualUsingResidualEquation1Q100K5", "GranularResidualUsingResidualEquation2Q100K5", "GranularResidualUsingResidualEquation3Q100K5",
      "GranularResidualUsingResidualEquation4Q100K5", "GranularResidualUsingResidualEquation5Q100K5", "GranularResidualUsingResidualEquation6Q100K5")
```

Once I have computed both the $K = 100$ and $K = 5$ samples I merge them together by year and create a single `GranularResiduals` data frame. I save a 3 series subset ($(Q = 100, K = 100)$ for both the year and year by industry mean computations as well as the $(Q = 100, K = 5)$ estimates using the year by industry means) from this consolidated granular residual data frame for Xavier to post on the web.

```
> GranularResiduals <- merge(GranularResidualsK100, GranularResidualsK5, by = "year")
> OnlineData <- GranularResiduals[, c("year", "GranularResidualUsingYearAndIndustryDemeaningQ100K100", "GranularResidualUsingYearDemeaningQ100K100",
      "GranularResidualUsingYearAndIndustryDemeaningQ100K5")]
> write.csv(OnlineData, file = paste(DATA_DIRECTORY, "OnlineData_10Oct2010.csv", sep = ""))
```

I then merge this consolidated data frame of granular residual estimates back onto the main `CompustatData` data frame and save the data to file both as a `Rdata` file as well as a `csv`

```
> CompustatData <- merge(CompustatData, GranularResiduals, by = "year")
> save(CompustatData, GranularResiduals, file = paste(DATA_DIRECTORY, "GranularResidualData_10Oct2010.Rdata", sep = ""))
> write.csv(CompustatData, file = paste(DATA_DIRECTORY, "GranularResidualData_10Oct2010.csv", sep = ""))
```

In addition to saving the complete granular residual data, I also save an abridged version of the data frame with fewer variables so that Farzad Saidi can investigate in more detail the stories behind some of the larger events.

```
> NarrativeData <- CompustatData[, c("year", "ticker", "companyName", "sicCode", "numberOfEmployees", "sales", "laggedNumberOfEmployees",
      "laggedSales", "changeInLogProductivity", "laggedRealGdp", "meanChangeInLogProductivityQ100GroupedByYear", "meanChangeInLogProductivityQ100GroupedByYearAndIndustry",
      "unwinsorizedDemeanedChangeInLogProductivityUsingMeanQ100GroupedByYear", "unwinsorizedDemeanedChangeInLogProductivityUsingMeanQ100GroupedByYearAndIndustry",
      "demeanedChangeInLogProductivityUsingMeanQ100GroupedByYear", "demeanedChangeInLogProductivityUsingMeanQ100GroupedByYearAndIndustry",
      "meanLaggedLogSalesQ100GroupedByYear", "meanLaggedLogSalesQ100GroupedByYearAndIndustry", "demeanedLaggedLogSalesUsingMeanQ100GroupedByYear",
      "demeanedLaggedLogSalesUsingMeanQ100GroupedByYearAndIndustry", "firmContributionToGranularResidualUsingYearDemeaningQ100", "firmContributionToGranularResidualUsingYearAndIndustryDemeaningQ100",
      "GranularResidualUsingYearDemeaningQ100K100", "GranularResidualUsingYearAndIndustryDemeaningQ100K100")]
> NarrativeData <- NarrativeData[order(NarrativeData$year, -NarrativeData$laggedSales), ]
> write.csv(NarrativeData, file = paste(DATA_DIRECTORY, "NarrativeData_10Oct2010.csv", sep = ""))
```

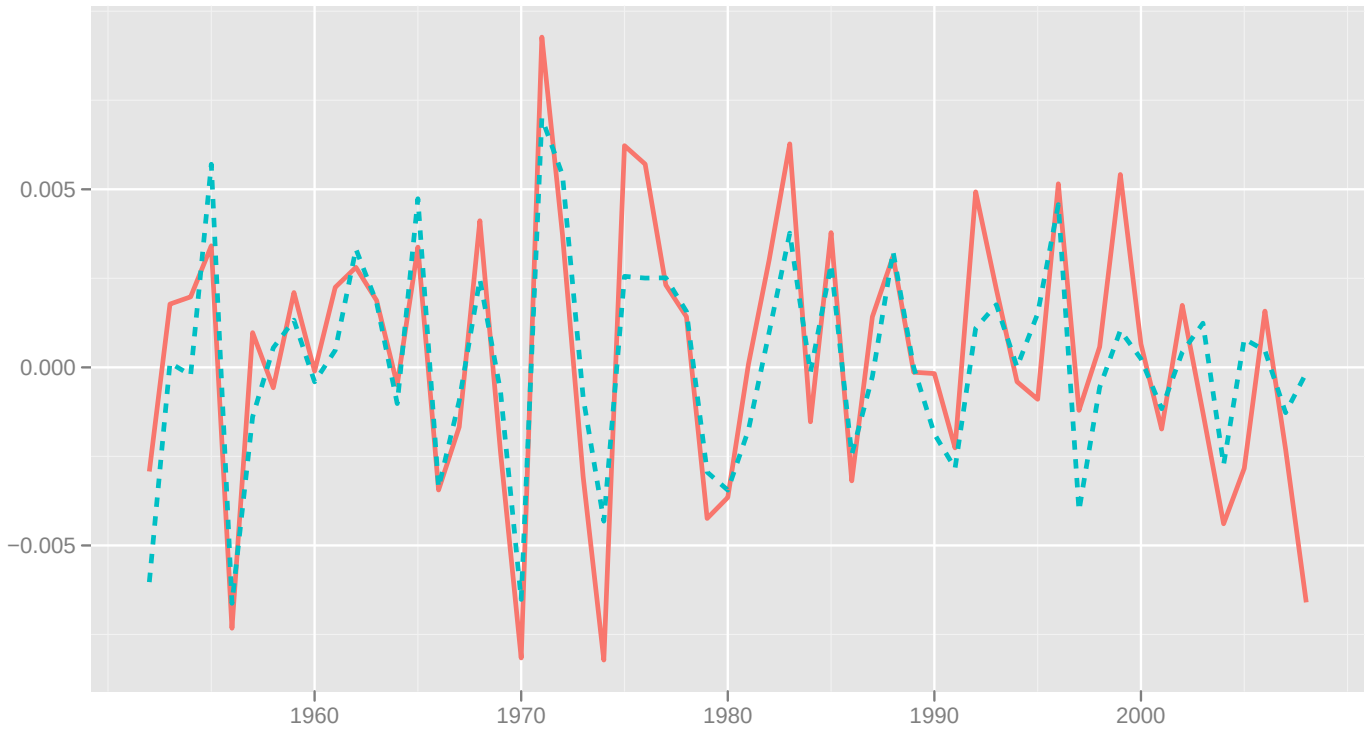

Figure 1: ***Q=100, K=100, Granular Residual Time Series****: This figure shows the granular residual for the* ($Q = 100, K = 100$) *specification using the productivity growth rates estimated by demeaning the variables at the year (orange, solid) and year by industry level (blue, dashed).*

Plot Granular Residuals

Now that I have computed the granular residuals, I create some plots and summary statistics to make sure everything looks right.

```
> pdf(paste(FIGURES_DIRECTORY, "TimeSeries_GranularResidualsUsingDemeanedDataAtBothYearAndIndustryYearLevels.pdf", sep = ""), height = 5,
      width = 9)
> PlotData <- GranularResiduals[, c("year", "GranularResidualUsingYearDemeaningQ100K100", "GranularResidualUsingYearAndIndustryDemeaningQ100K100")]
> names(PlotData) <- c("year", "Year Mean", "Year x Industry Mean")
> PlotData <- melt(PlotData, c("year"))
> p <- ggplot(PlotData, aes(x = year, y = value, group = variable, colour = variable, linetype = variable))
> p <- p + geom_path(size = 1)
> p <- p + opts(legend.position = "none")
> p <- p + ylab("") + xlab("")
> print(p)
> dev.off()
```

Compare Narrative Data to Earlier Version Sent to Farzad in June 2010

Finally, I conclude by comparing the current iteration of the data to the version I sent Farzad in June 2010. Below, I load in the earlier data and rename the variables to match the naming conventions in the code above.

```
> NarrativeData_18June2010 <- read.csv(file = paste(DATA_DIRECTORY, "NarrativeDataForFarzad_18June2010.csv", sep = ""), stringsAsFactors = FALSE)
> NarrativeData_18June2010 <- NarrativeData_18June2010[c(1:10, 13, 15, 26:30)]
> names(NarrativeData_18June2010) <- c("year", "industry", "id", "ticker", "companyName", "sicCode", "sales", "numberOfEmployees", "realGdp",
      "laggedSales", "productivity", "changeInLogProductivity", "laggedRealGdp", "firmContributionToGranularResidualUsingYearDemeaningQ100",
      "firmContributionToGranularResidualUsingYearAndIndustryDemeaningQ100", "GranularResidualUsingYearDemeaningQ100K100", "GranularResidualUsingYearAndIndustryDemeaningQ100K100")
> NarrativeData_18June2010 <- NarrativeData_18June2010[NarrativeData_18June2010$year %in% seq(1952, 2008), ]
```

Next, I compare the orderings in the June and current versions of the data. For each year, I plot the ticker symbols of the companies which are in the set difference of the 2 narrative data sets. These are the companies whose appearance or disappearance need to be explained by a change in the data processing or selection procedures.

There are 2 main differences between the current version of the narrative data and the version from June. First, the version from June includes financial firms whereas the current specification excludes firms from this industry. For example, the current version of the data excludes Citibank from the sample even though it is the 6^{th} largest firm sorted by lagged sales in the 2000's.

Second, the current version of the data drops firms which do not have valid data for the current and lagged sales volume and number of of employees prior to doing any rankings. The earlier version of the narrative data ranked the firms prior to investigating whether or not valid data existed for both the current and lagged periods. For example, the Great Atlantic

and Pacific Tea Co. (GAP) exists in the June version of the data in 1959 but not the current version. GAP does not report employee count data before 1959 but has sales volume data back to the beginning of the COMPUSTAT sample.

All of the new firms added in the current version are firms that would otherwise be ranked out of the top 100 as sorted by lagged sales, but because I now remove all of the financial firms and also clean out any firms with, say, missing lagged employee count data these firms now make the cut.

```
> NarrativeData_18June2010 <- NarrativeData_18June2010[order(NarrativeData_18June2010$year, -NarrativeData_18June2010$laggedSales), ]
> NarrativeData <- NarrativeData[order(NarrativeData$year, -NarrativeData$laggedSales), ]
> NarrativeData_18June2010$order <- seq(1, 100)
> NarrativeData$order <- seq(1, 100)
> CompaniesInJuneDataSetButNotCurrentDataSet <- data.frame(c())
> CompaniesInCurrentDataSetButNotJuneDataSet <- data.frame(c())
> for (YEAR in seq(1952, 2008)) {
      NarrativeData_18June2010_For1Year <- NarrativeData_18June2010[NarrativeData_18June2010$year == YEAR, ]
      NarrativeData_For1Year <- NarrativeData[NarrativeData$year == YEAR, ]
      CompaniesInJuneDataSetButNotCurrentDataSet <- rbind(CompaniesInJuneDataSetButNotCurrentDataSet, NarrativeData_18June2010_For1Year[!(NarrativeData_18June2010_For1Year$companyName %in%
          NarrativeData_For1Year$companyName), ])
      CompaniesInCurrentDataSetButNotJuneDataSet <- rbind(CompaniesInCurrentDataSetButNotJuneDataSet, NarrativeData_For1Year[!(NarrativeData_For1Year$companyName %in%
          NarrativeData_18June2010_For1Year$companyName), ])
  }
> CompaniesInJuneDataSetButNotCurrentDataSet <- CompaniesInJuneDataSetButNotCurrentDataSet[, c("year", "order", "ticker")]
> CompaniesInJuneDataSetButNotCurrentDataSet$variable <- "Unique to June"
> CompaniesInCurrentDataSetButNotJuneDataSet <- CompaniesInCurrentDataSetButNotJuneDataSet[, c("year", "order", "ticker")]
> CompaniesInCurrentDataSetButNotJuneDataSet$variable <- "Unique to Now"
> PlotData <- rbind(CompaniesInCurrentDataSetButNotJuneDataSet, CompaniesInJuneDataSetButNotCurrentDataSet)
> names(PlotData) <- c("year", "order", "value", "variable")
> PlotData$variable <- as.factor(PlotData$variable)
> pdf(paste(FIGURES_DIRECTORY, "ScatterPlot_FirmRankingsOverTimeInJuneAndCurrentData.pdf", sep = ""), height = 9, width = 11)
> p <- ggplot(PlotData, aes(x = year, y = order, colour = variable, label = value))
> p <- p + geom_text(size = 1.5)
> p <- p + ylab("Rank by Lagged Sales ($MM)") + xlab("Year")
> print(p)
> dev.off()
```

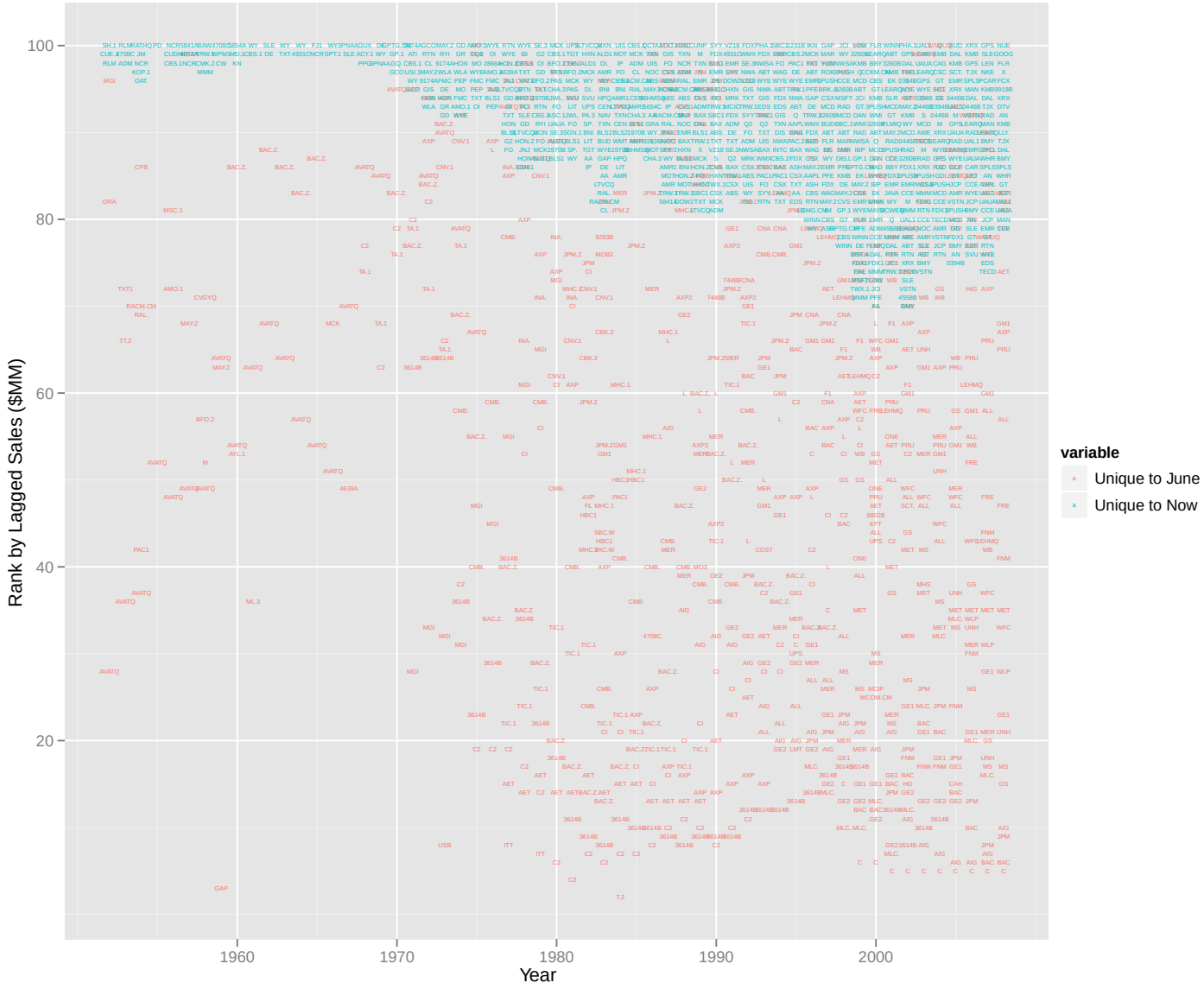

Figure 2: ***June vs. October 2010 Narrative Data****: This figure shows difference in firm rankings as sorted by lagged sales for the top* 100 *firms in the COMPUSTAT data between the narrative data in June and October 2010. For each year, I plot the ticker symbols of the companies which are in the set difference of the* 2 *narrative data sets. These are the companies whose appearance or disappearance need to be explained by a change in the data processing or selection procedures. There are* 2 *main differences between the current version of the narrative data and the version from June. First, the version from June includes financial firms whereas the current specification excludes firms from this industry. For example, the current version of the data excludes Citibank from the sample even though it is the* 6^{th} *largest firm sorted by lagged sales in the* 2000*'s. Second, the current version of the data drops firms which do not have valid data for the current and lagged sales volume and number of of employees prior to doing any rankings. The earlier version of the narrative data ranked the firms prior to investigating whether or not valid data existed for both the current and lagged periods. For example, the Great Atlantic and Pacific Tea Co. (GAP) exists in the June version of the data in* 1959 *but not the current version. GAP does not report employee count data before* 1959 *but has sales volume data back to the beginning of the COMPUSTAT sample.*