StrategicTrade

Empirical Market Microstructure

(2006, Oxford University Press)

Companion *Mathematica* notebook

Joel Hasbrouck

This notebook covers material related to the strategic trade models (Chapter 7).

```
Text @ Style["Notebook evaluated " <> DateString["DateTime"], "Subtitle"]
```

Notebook evaluated Monday 4 June 2007 20:23:25

Preliminaries

Invoke the multivariate normal package

```
SetDirectory[
  "c:/Active/Empirical Market Microstructure/Mathematica/Spring 2007"];
```

```
<< MVN.m
```

Notations

```
<< Notation`
```

The following commands define symbolizations that are convenient for labeling things.

```
Symbolize[Anything_Rule]; Symbolize[Anything_Rules];
Symbolize[Anything_Solution]; Symbolize[Anything_Solutions];
```

The following command allows the "expectation" operator to be entered as ESC E ESC.

```
AddInputAlias[ℰ, "E"]
```

```
Symbolize[σ²_]
```

```
Symbolize[μ_]
```

```
Symbolize[Σ_]
```

■ The single-period model

◻ *The informed trader's problem*

The informed trader conjectures that the MM uses a linear price adjustment rule:

```
pRule = p → y λ + μ;
```

where y is the total order flow:

```
yRule = y → u + x;
```

The informed trader's profits are:

```
πRule = π → (v - p) x;
```

Substituting in for the price conjecture and y:

```
π /. πRule /. pRule /. yRule
```

$$x\,(v - (u + x)\,\lambda - \mu)$$

The expected profits are Eπ:

```
Eπ = π /. πRule /. pRule /. yRule /. u → 0
```

$$x\,(v - x\,\lambda - \mu)$$

The informed trader maximizes expected profits by trading x:

```
xOpt = First @ Solve[∂x Eπ == 0, x]
```

$$\left\{x \to \frac{v - \mu}{2\,\lambda}\right\}$$

The second-order condition for the max is

```
∂x,x Eπ < 0
```

$$-2\,\lambda < 0$$

The market maker's problem

The MM conjectures that the informed trader's demand is linear in v:

```
xRule = x → α + v β
```

$x \to \alpha + v\,\beta$

Knowing the optimization process that the informed trader followed, the MM can solve for α and β:

```
xEqu = (x /. xRule) == (x /. xOpt)
```

$$\alpha + v\,\beta == \frac{v - \mu}{2\,\lambda}$$

for all *v*. This implies:

```
r = Reduce[ForAll[v, True, xEqu], {α, β}, Reals]
```

$$\left(\lambda < 0 \,\&\&\, \alpha == -\frac{\mu}{2\,\lambda} \,\&\&\, \beta == \frac{1}{2\,\lambda}\right) \,||\, \left(\lambda > 0 \,\&\&\, \alpha == -\frac{\mu}{2\,\lambda} \,\&\&\, \beta == \frac{1}{2\,\lambda}\right)$$

```
xSolutions = ToRules[r[[2, {2, 3}]]]
```

$$\left\{\alpha \to -\frac{\mu}{2\,\lambda},\ \beta \to \frac{1}{2\,\lambda}\right\}$$

Now the MM must figure out $E[V \mid Y = y]$. First, consider the joint distribution of v and u:

```
uvDist = MVN[{{p0}, {0}}, {{Σ0, 0}, {0, σu^2}}, {v, u}]
```

$$\begin{pmatrix} v \\ u \end{pmatrix} \sim \mathcal{N}\left(\begin{pmatrix} p_0 \\ 0 \end{pmatrix}, \begin{pmatrix} \Sigma_0 & 0 \\ 0 & \sigma_u^2 \end{pmatrix}\right)$$

Now consider the joint distribution of v and y where y =

```
y /. yRule /. xRule
```

$u + \alpha + v\,\beta$

So

```
MakeLinearForm[uvDist, {v, y /. yRule /. xRule}]
```

$$\begin{pmatrix} v \\ u + \alpha + v\,\beta \end{pmatrix} \sim \mathcal{N}\left(\begin{pmatrix} p_0 \\ \alpha + \beta\,p_0 \end{pmatrix}, \begin{pmatrix} \Sigma_0 & \beta\,\Sigma_0 \\ \beta\,\Sigma_0 & \beta^2\,\Sigma_0 + \sigma_u^2 \end{pmatrix}\right)$$

Relabeling:

```
vyDist = SetLabel[%, {v, y}]
```

$$\begin{pmatrix} v \\ y \end{pmatrix} \sim \mathcal{N}\left(\begin{pmatrix} p_0 \\ \alpha + \beta\, p_0 \end{pmatrix}, \begin{pmatrix} \Sigma_0 & \beta\,\Sigma_0 \\ \beta\,\Sigma_0 & \beta^2\,\Sigma_0 + \sigma_u^2 \end{pmatrix}\right)$$

Now consider the conditional distribution $v \mid y$:

```
vConditionalDist = MVNConditional[vyDist, v, y]
```

$$v \sim \mathcal{N}\left(p_0 + \frac{\beta\,\Sigma_0\,(y - \alpha - \beta\,p_0)}{\beta^2\,\Sigma_0 + \sigma_u^2},\ \Sigma_0 - \frac{\beta^2\,\Sigma_0^2}{\beta^2\,\Sigma_0 + \sigma_u^2}\right)$$

Market efficiency requires that the mean of this distribution be equal to p.

```
GetMean[vConditionalDist] == (p /. pRule)
```

$$p_0 + \frac{\beta\,\Sigma_0\,(y - \alpha - \beta\,p_0)}{\beta^2\,\Sigma_0 + \sigma_u^2} == y\,\lambda + \mu$$

for all values of y.

```
r = Reduce[ForAll[y, True, %], {μ, λ}]
```

$$\beta^2\,\Sigma_0 + \sigma_u^2 \neq 0\ \&\&\ \mu == \frac{-\alpha\,\beta\,\Sigma_0 + \sigma_u^2\,p_0}{\beta^2\,\Sigma_0 + \sigma_u^2}\ \&\&\ \lambda == \frac{\beta\,\Sigma_0}{\beta^2\,\Sigma_0 + \sigma_u^2}$$

```
pSolutions = ToRules[r[[{2, 3}]]]
```

$$\left\{\mu \to \frac{-\alpha\,\beta\,\Sigma_0 + \sigma_u^2\,p_0}{\beta^2\,\Sigma_0 + \sigma_u^2},\ \lambda \to \frac{\beta\,\Sigma_0}{\beta^2\,\Sigma_0 + \sigma_u^2}\right\}$$

Collecting these results...

```
EquationSet = (Equal @@ #1 &) /@ Join[pSolutions, xSolutions]
```

$$\left\{\mu == \frac{-\alpha\,\beta\,\Sigma_0 + \sigma_u^2\,p_0}{\beta^2\,\Sigma_0 + \sigma_u^2},\ \lambda == \frac{\beta\,\Sigma_0}{\beta^2\,\Sigma_0 + \sigma_u^2},\ \alpha == -\frac{\mu}{2\,\lambda},\ \beta == \frac{1}{2\,\lambda}\right\}$$

... and solving for the parameters in terms of the fundamental variables.

```
ModelSolutions = Solve[EquationSet, {μ, λ, α, β}]; ModelSolutions // TableForm
```

$$\begin{array}{llll} \alpha \to -\frac{\sqrt{\sigma_u^2}\,p_0}{\sqrt{\Sigma_0}} & \mu \to p_0 & \lambda \to \frac{\sqrt{\Sigma_0}}{2\sqrt{\sigma_u^2}} & \beta \to \frac{\sqrt{\sigma_u^2}}{\sqrt{\Sigma_0}} \\ \alpha \to \frac{\sqrt{\sigma_u^2}\,p_0}{\sqrt{\Sigma_0}} & \mu \to p_0 & \lambda \to -\frac{\sqrt{\Sigma_0}}{2\sqrt{\sigma_u^2}} & \beta \to -\frac{\sqrt{\sigma_u^2}}{\sqrt{\Sigma_0}} \end{array}$$

Only first solution has positive λ and β, so:

```
Model_Solutions = First @ Model_Solutions ; TableForm @ Model_Solutions
```

$$\alpha \to -\frac{\sqrt{\sigma_u^2}\ p_0}{\sqrt{\Sigma_0}}$$

$$\mu \to p_0$$

$$\lambda \to \frac{\sqrt{\Sigma_0}}{2\sqrt{\sigma_u^2}}$$

$$\beta \to \frac{\sqrt{\sigma_u^2}}{\sqrt{\Sigma_0}}$$

Note:

```
Simplify[y_Rule /. x_Rule /. Model_Solutions]
```

$$y \to u + \frac{v\sqrt{\sigma_u^2}}{\sqrt{\Sigma_0}} - \frac{\sqrt{\sigma_u^2}\ p_0}{\sqrt{\Sigma_0}}$$

Properties of the solution

The informed trader's expected profits are:

```
Simplify[PowerExpand[Eπ /. x_Rule //. Model_Solutions]]
```

$$\frac{\sqrt{\sigma_u^2}\ (v - p_0)^2}{2\sqrt{\Sigma_0}}$$

The informed trader's demand is

```
x_Rule /. Model_Solutions // Simplify
```

$$x \to \frac{\sqrt{\sigma_u^2}\ (v - p_0)}{\sqrt{\Sigma_0}}$$

How much of the private information is impounded in the price?

```
GetVariance[vConditionalDist] // Simplify
```

$$\frac{\Sigma_0\ \sigma_u^2}{\beta^2\ \Sigma_0 + \sigma_u^2}$$

Or, in terms of the input parameters:

```
Simplify[% //. Model_Solutions]
```

$$\frac{\Sigma_0}{2}$$

Problems based on the single-period model

Exercise 7.1 (Informative noise traders)

As in the basic problem:

```
pRule = p → y λ + μ;
yRule = y → u + x;
πRule = π → (v - p) x;
```

The informed trader's profits are:

```
π /. πRule /. pRule /. yRule
```

$$x\,(v - (u + x)\,\lambda - \mu)$$

At this point we diverge from the basic model because u and v are correlated. The projection the informed trader makes is:

```
uvDist = MVN[{{p0}, {0}}, {{Σ0, σuv}, {σuv, σu^2}}, {v, u}]
```

$$\begin{pmatrix} v \\ u \end{pmatrix} \sim \mathcal{N}\left(\begin{pmatrix} p_0 \\ 0 \end{pmatrix}, \begin{pmatrix} \Sigma_0 & \sigma_{uv} \\ \sigma_{uv} & \sigma_u^2 \end{pmatrix}\right)$$

```
uConditionalDist = MVNConditional[uvDist, u, v]
```

$$u \sim \mathcal{N}\left(\frac{(v - p_0)\,\sigma_{uv}}{\Sigma_0}, \sigma_u^2 - \frac{\sigma_{uv}^2}{\Sigma_0}\right)$$

The informed trader's expected profits (conditional on v) are:

```
Eπ = π /. πRule /. pRule /. yRule /. u → GetMean[uConditionalDist]
```

$$x\left(v - \mu - \lambda\left(x + \frac{(v - p_0)\,\sigma_{uv}}{\Sigma_0}\right)\right)$$

The informed trader's optimal trade is:

```
xOpt = First @ Simplify[Solve[∂x Eπ == 0, x]]
```

$$\left\{x \to \frac{(v - \mu)\,\Sigma_0 + \lambda\,(-v + p_0)\,\sigma_{uv}}{2\,\lambda\,\Sigma_0}\right\}$$

The MM conjectures that the informed trader's demand is linear in v (as above), and must figure out E[v | y]:

```
xRule = x → α + v β;
xEqu = (x /. xRule) == (x /. xOpt)
```

$$\alpha + v\,\beta == \frac{(v-\mu)\,\Sigma_0 + \lambda\,(-v + p_0)\,\sigma_{uv}}{2\,\lambda\,\Sigma_0}$$

```
xSolutions = Reduce[ ∀v xEqu && Σ0 > 0 && λ > 0, {α, β}, Reals]
```

$$\Sigma_0 > 0 \;\&\&\; \lambda > 0 \;\&\&\; \alpha == \frac{-\mu\,\Sigma_0 + \lambda\,p_0\,\sigma_{uv}}{2\,\lambda\,\Sigma_0} \;\&\&\; \beta == \frac{\Sigma_0 - \lambda\,\sigma_{uv}}{2\,\lambda\,\Sigma_0}$$

```
xSolutions = Simplify @ ToRules @ Take[xSolutions, -2]
```

$$\left\{\alpha \to \frac{1}{2}\left(-\frac{\mu}{\lambda} + \frac{p_0\,\sigma_{uv}}{\Sigma_0}\right),\; \beta \to \frac{\Sigma_0 - \lambda\,\sigma_{uv}}{2\,\lambda\,\Sigma_0}\right\}$$

Now the MM must compute $E[v \mid y]$:

```
MakeLinearForm[uvDist, {v, y /. yRule /. xRule}]
```

$$\begin{pmatrix} v \\ u + \alpha + v\,\beta \end{pmatrix} \sim \mathcal{N}\left(\begin{pmatrix} p_0 \\ \alpha + \beta\,p_0 \end{pmatrix}, \begin{pmatrix} \Sigma_0 & \beta\,\Sigma_0 + \sigma_{uv} \\ \beta\,\Sigma_0 + \sigma_{uv} & \sigma_u^2 + \beta\,\sigma_{uv} + \beta\,(\beta\,\Sigma_0 + \sigma_{uv}) \end{pmatrix}\right)$$

```
vyDist = SetLabel[%, {v, y}]
```

$$\begin{pmatrix} v \\ y \end{pmatrix} \sim \mathcal{N}\left(\begin{pmatrix} p_0 \\ \alpha + \beta\,p_0 \end{pmatrix}, \begin{pmatrix} \Sigma_0 & \beta\,\Sigma_0 + \sigma_{uv} \\ \beta\,\Sigma_0 + \sigma_{uv} & \sigma_u^2 + \beta\,\sigma_{uv} + \beta\,(\beta\,\Sigma_0 + \sigma_{uv}) \end{pmatrix}\right)$$

```
vConditionalDist = MVNConditional[vyDist, v, y] // Simplify
```

$$v \sim \mathcal{N}\left(\frac{(y-\alpha)\,(\beta\,\Sigma_0 + \sigma_{uv}) + p_0\,(\sigma_u^2 + \beta\,\sigma_{uv})}{\beta^2\,\Sigma_0 + \sigma_u^2 + 2\,\beta\,\sigma_{uv}}, \frac{\Sigma_0\,\sigma_u^2 - \sigma_{uv}^2}{\beta^2\,\Sigma_0 + \sigma_u^2 + 2\,\beta\,\sigma_{uv}}\right)$$

Market efficiency:

```
pEqu = GetMean[vConditionalDist] == (p /. pRule)
```

$$\frac{(y-\alpha)\,(\beta\,\Sigma_0 + \sigma_{uv}) + p_0\,(\sigma_u^2 + \beta\,\sigma_{uv})}{\beta^2\,\Sigma_0 + \sigma_u^2 + 2\,\beta\,\sigma_{uv}} == y\,\lambda + \mu$$

```
r = Reduce[ForAll[y, True, pEqu], {μ, λ}]
```

$$\beta^2\,\Sigma_0 + \sigma_u^2 + 2\,\beta\,\sigma_{uv} \neq 0 \;\&\&\; \mu == \frac{-\alpha\,\beta\,\Sigma_0 + \sigma_u^2\,p_0 - \alpha\,\sigma_{uv} + \beta\,p_0\,\sigma_{uv}}{\beta^2\,\Sigma_0 + \sigma_u^2 + 2\,\beta\,\sigma_{uv}} \;\&\&\; \lambda == \frac{\beta\,\Sigma_0 + \sigma_{uv}}{\beta^2\,\Sigma_0 + \sigma_u^2 + 2\,\beta\,\sigma_{uv}}$$

```
p_Solutions = ToRules[Take[r, -2]]
```

$$\left\{\mu \to \frac{-\alpha\,\beta\,\Sigma_0 + \sigma_u^2\,p_0 - \alpha\,\sigma_{uv} + \beta\,p_0\,\sigma_{uv}}{\beta^2\,\Sigma_0 + \sigma_u^2 + 2\,\beta\,\sigma_{uv}},\ \lambda \to \frac{\beta\,\Sigma_0 + \sigma_{uv}}{\beta^2\,\Sigma_0 + \sigma_u^2 + 2\,\beta\,\sigma_{uv}}\right\}$$

Collecting the results and solving:

```
EquSet = Apply[Equal, Join[p_Solutions, x_Solutions], 1]
```

$$\left\{\mu == \frac{-\alpha\,\beta\,\Sigma_0 + \sigma_u^2\,p_0 - \alpha\,\sigma_{uv} + \beta\,p_0\,\sigma_{uv}}{\beta^2\,\Sigma_0 + \sigma_u^2 + 2\,\beta\,\sigma_{uv}},\ \lambda == \frac{\beta\,\Sigma_0 + \sigma_{uv}}{\beta^2\,\Sigma_0 + \sigma_u^2 + 2\,\beta\,\sigma_{uv}},\ \alpha == \frac{1}{2}\left(-\frac{\mu}{\lambda} + \frac{p_0\,\sigma_{uv}}{\Sigma_0}\right),\ \beta == \frac{\Sigma_0 - \lambda\,\sigma_{uv}}{2\,\lambda\,\Sigma_0}\right\}$$

```
Model_Solutions = Simplify[Solve[EquSet, {μ, λ, α, β}], {σ_u^2 > 0, Σ_0 > 0}];
Model_Solutions // Transpose // TableForm
```

$$\alpha \to \frac{p_0\left(\sigma_{uv} - \sqrt{4\,\Sigma_0\,\sigma_u^2 - 3\,\sigma_{uv}^2}\right)}{2\,\Sigma_0} \qquad \alpha \to \frac{p_0\left(\sigma_{uv} + \sqrt{4\,\Sigma_0\,\sigma_u^2 - 3\,\sigma_{uv}^2}\right)}{2\,\Sigma_0}$$

$$\mu \to p_0 \qquad \mu \to p_0$$

$$\lambda \to \frac{\Sigma_0\left(\sigma_{uv} + \sqrt{4\,\Sigma_0\,\sigma_u^2 - 3\,\sigma_{uv}^2}\right)}{4\,\Sigma_0\,\sigma_u^2 - 3\,\sigma_{uv}^2 + \sigma_{uv}\sqrt{4\,\Sigma_0\,\sigma_u^2 - 3\,\sigma_{uv}^2}} \qquad \lambda \to \frac{\Sigma_0\left(-\sigma_{uv} + \sqrt{4\,\Sigma_0\,\sigma_u^2 - 3\,\sigma_{uv}^2}\right)}{-4\,\Sigma_0\,\sigma_u^2 + 3\,\sigma_{uv}^2 + \sigma_{uv}\sqrt{4\,\Sigma_0\,\sigma_u^2 - 3\,\sigma_{uv}^2}}$$

$$\beta \to \frac{-\sigma_{uv} + \sqrt{4\,\Sigma_0\,\sigma_u^2 - 3\,\sigma_{uv}^2}}{2\,\Sigma_0} \qquad \beta \to -\frac{\sigma_{uv} + \sqrt{4\,\Sigma_0\,\sigma_u^2 - 3\,\sigma_{uv}^2}}{2\,\Sigma_0}$$

Only the first solution can have $\beta > 0$, So:

```
Model_Solutions = Model_Solutions[[1]];
```

When $\sigma_{uv} = 0$, this reduces to the original solutions:

```
Simplify[Model_Solutions /. σ_uv → 0, {Σ_0 > 0, σ_u^2 > 0}]
```

$$\left\{\alpha \to -\frac{\sigma_u^2\,p_0}{\sqrt{\Sigma_0\,\sigma_u^2}},\ \mu \to p_0,\ \lambda \to \frac{\Sigma_0}{2\sqrt{\Sigma_0\,\sigma_u^2}},\ \beta \to \sqrt{\frac{\sigma_u^2}{\Sigma_0}}\right\}$$

With perfect correlation...

```
AltSolution = Simplify[Model_Solutions /. σ_uv → Sqrt[σ_u^2 Σ_0], {Σ_0 > 0, σ_u^2 > 0}]
```

$$\left\{\alpha \to 0,\ \mu \to p_0,\ \lambda \to \sqrt{\frac{\Sigma_0}{\sigma_u^2}},\ \beta \to 0\right\}$$

Since $\beta = 0$, the informed trader doesn't trade at all. The uninformed trade, though, is linear in v:

$$u_{Rule} = u \to \frac{(v - p_0)\sqrt{\sigma_u^2\,\Sigma_0}}{\Sigma_0};$$

Under these conditions, the market clearing price becomes $p = p_0 + \lambda\, y = v$.

```
Simplify[pRule /. AltSolution /. yRule /. x → 0 /. uRule, {Σ0 > 0, σu^2 > 0}]
```

$p \to v$

Also, the conditional variance is:

```
GetVariance[vConditionalDist] //. ModelSolutions /. σuv → Sqrt[σu^2 Σ0]
```

0

Exercise 7.2 (Informed trader gets a signal)

The informed trader in the basic model has perfect information about *v*. Consider the case where she only gets a signal *s* about *v*. That is, $s = v + \epsilon$ where $\epsilon \sim N[0,\ \sigma_\epsilon^2]$, independent of v. Solve the model by proceeding as in the basic case. Solve the informed trader's problem; solve the MM's problem; solve for the model parameters $(\alpha,\ \beta,\ \mu,\ \lambda)$ in terms of the inputs, $\sigma_u^2,\ \ \Sigma_0,$ and σ_ϵ^2. Interpret your results. Verify that when $\sigma_\epsilon^2 = 0$, you get the original model solutions.

Solution

```
pRule = p → y λ + μ;
yRule = y → u + x;
πRule = π → (v - p) x;
```

Informed trader's profits:

```
π /. πRule /. pRule /. yRule
```

$x\,(v - (u + x)\,\lambda - \mu)$

The informed trader gets the signal s:

```
sRule = s → v + ϵ;
```

There are three random variables in this problem:

```
vϵuDistribution = MVN[{p0, 0, 0}, {{Σ0, 0, 0}, {0, σϵ^2, 0}, {0, 0, σu^2}}, {v, ϵ, u}]
```

$$\begin{pmatrix} v \\ \epsilon \\ u \end{pmatrix} \sim \mathcal{N}\left(\begin{pmatrix} p_0 \\ 0 \\ 0 \end{pmatrix}, \begin{pmatrix} \Sigma_0 & 0 & 0 \\ 0 & \sigma_\epsilon^2 & 0 \\ 0 & 0 & \sigma_u^2 \end{pmatrix}\right)$$

We can rework this into a distribution for v, s, u:

```
MakeLinearForm[vϵuDistribution, {v, s /. sRule, u}]
```

$$\begin{pmatrix} v \\ v+\epsilon \\ u \end{pmatrix} \sim \mathcal{N}\left(\begin{pmatrix} p_0 \\ p_0 \\ 0 \end{pmatrix}, \begin{pmatrix} \Sigma_0 & \Sigma_0 & 0 \\ \Sigma_0 & \Sigma_0+\sigma_\epsilon^2 & 0 \\ 0 & 0 & \sigma_u^2 \end{pmatrix}\right)$$

```
vsuDistribution = SetLabel[%, {v, s, u}]
```

$$\begin{pmatrix} v \\ s \\ u \end{pmatrix} \sim \mathcal{N}\left(\begin{pmatrix} p_0 \\ p_0 \\ 0 \end{pmatrix}, \begin{pmatrix} \Sigma_0 & \Sigma_0 & 0 \\ \Sigma_0 & \Sigma_0+\sigma_\epsilon^2 & 0 \\ 0 & 0 & \sigma_u^2 \end{pmatrix}\right)$$

The informed trader forms the conditional distribution of v based on his signal:

```
vConditionalDistInf = MVNConditional[vsuDistribution, v, s]
```

$$v \sim \mathcal{N}\left(\frac{\Sigma_0\,(s-p_0)}{\Sigma_0+\sigma_\epsilon^2}+p_0,\ \Sigma_0-\frac{\Sigma_0^2}{\Sigma_0+\sigma_\epsilon^2}\right)$$

The expected profits are developed from:

```
π /. πRule /. pRule /. yRule
```

$$x\,(v-(u+x)\,\lambda-\mu)$$

... substituting in the conditional mean for v:

```
Eπ = π /. πRule /. pRule /. yRule /. u → 0 /. v → GetMean[vConditionalDistInf]
```

$$x\left(-x\,\lambda-\mu+\frac{\Sigma_0\,(s-p_0)}{\Sigma_0+\sigma_\epsilon^2}+p_0\right)$$

The informed trader maximizes expected profits by trading *x*:

```
xOpt = First @ Simplify[Solve[∂x Eπ == 0, x]]
```

$$\left\{x\to\frac{s\,\Sigma_0-\mu\,(\Sigma_0+\sigma_\epsilon^2)+\sigma_\epsilon^2\,p_0}{2\,\lambda\,(\Sigma_0+\sigma_\epsilon^2)}\right\}$$

The MM conjectures that the informed trader's demand is linear in s:

```
xRule = x → α + s β;
```

Knowing the optimization process that the informed trader followed, the MM can solve for α and β:

```
xEqu = (x /. xRule) == (x /. xOpt)
```

$$\alpha + s\,\beta == \frac{s\,\Sigma_0 - \mu\,\left(\Sigma_0 + \sigma_\epsilon^2\right) + \sigma_\epsilon^2\,p_0}{2\,\lambda\,\left(\Sigma_0 + \sigma_\epsilon^2\right)}$$

```
xSolutions = Reduce[∀s xEqu && Σ0 > 0 && σϵ^2 > 0 && λ > 0 && s ≠ 0, {α, β}, Reals]
```

$$\sigma_\epsilon^2 > 0\ \&\&\ \Sigma_0 > 0\ \&\&\ \lambda > 0\ \&\&\ \left(\left(s < 0\ \&\&\ \alpha == \frac{-\mu\,\Sigma_0 - \mu\,\sigma_\epsilon^2 + \sigma_\epsilon^2\,p_0}{2\,\lambda\,\Sigma_0 + 2\,\lambda\,\sigma_\epsilon^2}\ \&\&\ \beta == \frac{\Sigma_0}{2\,\lambda\,\Sigma_0 + 2\,\lambda\,\sigma_\epsilon^2}\right) \,||\, \left(s > 0\ \&\&\ \alpha == \frac{-\mu\,\Sigma_0 - \mu\,\sigma_\epsilon^2 + \sigma_\epsilon^2\,p_0}{2\,\lambda\,\Sigma_0 + 2\,\lambda\,\sigma_\epsilon^2}\ \&\&\ \beta == \frac{\Sigma_0}{2\,\lambda\,\Sigma_0 + 2\,\lambda\,\sigma_\epsilon^2}\right)\right)$$

```
xSolutions = Simplify @ ToRules @ Take[xSolutions[[4, 1]], -2]
```

$$\left\{\alpha \to \frac{-\mu\,\left(\Sigma_0 + \sigma_\epsilon^2\right) + \sigma_\epsilon^2\,p_0}{2\,\lambda\,\left(\Sigma_0 + \sigma_\epsilon^2\right)},\ \beta \to \frac{\Sigma_0}{2\,\lambda\,\Sigma_0 + 2\,\lambda\,\sigma_\epsilon^2}\right\}$$

Now the MM must figure out $E[V \mid Y = y]$. This is a little more involved than in the original problem because the informed trader's demand is conditioned on s. The joint distribution of v and $y = u + \alpha + s\beta$ is:

```
MakeLinearForm[vsuDistribution, {v, y /. yRule /. xRule}]
```

$$\begin{pmatrix} v \\ u + \alpha + s\,\beta \end{pmatrix} \sim \mathcal{N}\left(\begin{pmatrix} p_0 \\ \alpha + \beta\,p_0 \end{pmatrix}, \begin{pmatrix} \Sigma_0 & \beta\,\Sigma_0 \\ \beta\,\Sigma_0 & \sigma_u^2 + \beta^2\,\left(\Sigma_0 + \sigma_\epsilon^2\right) \end{pmatrix}\right)$$

... and relabeling:

```
vyDistribution = SetLabel[%, {v, y}]
```

$$\begin{pmatrix} v \\ y \end{pmatrix} \sim \mathcal{N}\left(\begin{pmatrix} p_0 \\ \alpha + \beta\,p_0 \end{pmatrix}, \begin{pmatrix} \Sigma_0 & \beta\,\Sigma_0 \\ \beta\,\Sigma_0 & \sigma_u^2 + \beta^2\,\left(\Sigma_0 + \sigma_\epsilon^2\right) \end{pmatrix}\right)$$

So the distribution of v (conditional on y) is:

```
vConditionalDistributionMM = MVNConditional[vyDistribution, v, y]
```

$$v \sim \mathcal{N}\left(p_0 + \frac{\beta\,\Sigma_0\,(y - \alpha - \beta\,p_0)}{\sigma_u^2 + \beta^2\,\left(\Sigma_0 + \sigma_\epsilon^2\right)},\ \Sigma_0 - \frac{\beta^2\,\Sigma_0^2}{\sigma_u^2 + \beta^2\,\left(\Sigma_0 + \sigma_\epsilon^2\right)}\right)$$

Market efficiency requires

```
pEqu = GetMean[vConditionalDistributionMM] == (p /. pRule)
```

$$p_0 + \frac{\beta\,\Sigma_0\,(y - \alpha - \beta\,p_0)}{\sigma_u^2 + \beta^2\,\left(\Sigma_0 + \sigma_\epsilon^2\right)} == y\,\lambda + \mu$$

Solving:

```
pSolutions = Reduce[ ∀y pEqu && Σ0 > 0 && σϵ^2 > 0  && σu^2 > 0, {μ, λ}, Reals]
```

$$\sigma_\epsilon^2 > 0 \,\&\&\, \Sigma_0 > 0 \,\&\&\, \sigma_u^2 > 0 \,\&\&\, \mu == \frac{-\alpha\,\beta\,\Sigma_0 + \sigma_u^2\,p_0 + \beta^2\,\sigma_\epsilon^2\,p_0}{\beta^2\,\Sigma_0 + \sigma_u^2 + \beta^2\,\sigma_\epsilon^2} \,\&\&\, \lambda == \frac{\beta\,\Sigma_0}{\beta^2\,\Sigma_0 + \sigma_u^2 + \beta^2\,\sigma_\epsilon^2}$$

```
pSolutions = Simplify @ ToRules @ Take[pSolutions, -2]
```

$$\left\{\mu \to \frac{-\alpha\,\beta\,\Sigma_0 + \left(\sigma_u^2 + \beta^2\,\sigma_\epsilon^2\right)\,p_0}{\sigma_u^2 + \beta^2\,\left(\Sigma_0 + \sigma_\epsilon^2\right)},\ \lambda \to \frac{\beta\,\Sigma_0}{\sigma_u^2 + \beta^2\,\left(\Sigma_0 + \sigma_\epsilon^2\right)}\right\}$$

Collecting the results and solving:

```
EquationSet = Apply[Equal, Join[pSolutions, xSolutions], 1]
```

$$\left\{\mu == \frac{-\alpha\,\beta\,\Sigma_0 + \left(\sigma_u^2 + \beta^2\,\sigma_\epsilon^2\right)\,p_0}{\sigma_u^2 + \beta^2\,\left(\Sigma_0 + \sigma_\epsilon^2\right)},\ \lambda == \frac{\beta\,\Sigma_0}{\sigma_u^2 + \beta^2\,\left(\Sigma_0 + \sigma_\epsilon^2\right)},\ \alpha == \frac{-\mu\,\left(\Sigma_0 + \sigma_\epsilon^2\right) + \sigma_\epsilon^2\,p_0}{2\,\lambda\,\left(\Sigma_0 + \sigma_\epsilon^2\right)},\ \beta == \frac{\Sigma_0}{2\,\lambda\,\Sigma_0 + 2\,\lambda\,\sigma_\epsilon^2}\right\}$$

```
ModelSolutions =
 Simplify[Solve[EquationSet, {μ, λ, α, β}], {Σ0 > 0, σu^2 > 0, σϵ^2 > 0}] // First
```

$$\left\{\alpha \to -\sqrt{\frac{\sigma_u^2}{\Sigma_0 + \sigma_\epsilon^2}}\ p_0,\ \mu \to p_0,\ \lambda \to \frac{\Sigma_0}{2\,\sqrt{\sigma_u^2\,\left(\Sigma_0 + \sigma_\epsilon^2\right)}},\ \beta \to \sqrt{\frac{\sigma_u^2}{\Sigma_0 + \sigma_\epsilon^2}}\right\}$$

To recover the original solutions:

```
ModelSolutions /. σϵ^2 → 0
```

$$\left\{\alpha \to -\sqrt{\frac{\sigma_u^2}{\Sigma_0}}\ p_0,\ \mu \to p_0,\ \lambda \to \frac{\Sigma_0}{2\,\sqrt{\Sigma_0\,\sigma_u^2}},\ \beta \to \sqrt{\frac{\sigma_u^2}{\Sigma_0}}\right\}$$

Given the price (or equivalently the total order flow), the variance of v is:

```
GetVariance[vConditionalDistributionMM]
```

$$\Sigma_0 - \frac{\beta^2\,\Sigma_0^2}{\sigma_u^2 + \beta^2\,\left(\Sigma_0 + \sigma_\epsilon^2\right)}$$

As we degrade the signal (increase σ_ϵ^2), the conditional variance approaches the unconditional variance.

Exercise 7.3 (Broker piggy-backs on informed trader)

Solution

```
pRule = p → y λ + μ;
```

But now the order flow includes the broker's order flow γx.

```
yRule = y → u + x (1 + γ);
```

The informed trader's profits are:

```
πRule = π → (v - p) x;
```

Substituting in for the price conjecture and *y*:

```
π /. πRule /. pRule /. yRule
```

$$x\,(v - (u + x\,(1+\gamma))\,\lambda - \mu)$$

The expected profits (conditional on *v*) are Eπ:

```
Eπ = Simplify[π /. πRule /. pRule /. yRule /. u → 0]
```

$$x\,(v - x\,(1+\gamma)\,\lambda - \mu)$$

The optimal quantity is:

```
xOpt = First @ Solve[∂x Eπ == 0, x]
```

$$\left\{x \to \frac{v-\mu}{2\,(1+\gamma)\,\lambda}\right\}$$

```
xRule = x → α + v β;
xEqu = (x /. xRule) == (x /. xOpt)
```

$$\alpha + v\,\beta == \frac{v-\mu}{2\,(1+\gamma)\,\lambda}$$

Solving:

```
xSolutions = Reduce[ ∀v xEqu && λ > 0 && γ > 0 , {α, β}, Reals]
```

$$\lambda > 0\ \&\&\ \gamma > 0\ \&\&\ \alpha == -\frac{\mu}{2\,\lambda + 2\,\gamma\,\lambda}\ \&\&\ \beta == \frac{1}{2\,\lambda + 2\,\gamma\,\lambda}$$

```
xSolutions = Simplify @ ToRules @ Take[xSolutions, -2]
```

$$\left\{\alpha \to -\frac{\mu}{2\,\lambda + 2\,\gamma\,\lambda},\ \beta \to \frac{1}{2\,\lambda + 2\,\gamma\,\lambda}\right\}$$

As always, the MM must figure out $E[v \mid y]$. Starting from the original joint distribution:

```
uvDist = MVN[{{p0}, {0}}, {{Σ0, 0}, {0, σu^2}}, {v, u}]
```

$$\begin{pmatrix} v \\ u \end{pmatrix} \sim \mathcal{N}\left(\begin{pmatrix} p_0 \\ 0 \end{pmatrix}, \begin{pmatrix} \Sigma_0 & 0 \\ 0 & \sigma_u^2 \end{pmatrix}\right)$$

Now, though, we have a more complicated form for y : $y = u + (\alpha + v\,\beta)\,(1 + \gamma)$. The joint distribution of v and y is:

```
MakeLinearForm[uvDist, {v, y /. yRule /. xRule}]
```

$$\begin{pmatrix} v \\ u + (\alpha + v\,\beta)\,(1+\gamma) \end{pmatrix} \sim \mathcal{N}\left(\begin{pmatrix} p_0 \\ (\alpha + v\,\beta)\,(1+\gamma) - v\,(\beta + \beta\,\gamma) + (\beta + \beta\,\gamma)\,p_0 \end{pmatrix}, \begin{pmatrix} \Sigma_0 & (\beta + \beta\,\gamma)\,\Sigma_0 \\ (\beta + \beta\,\gamma)\,\Sigma_0 & (\beta + \beta\,\gamma)^2\,\Sigma_0 + \sigma_u^2 \end{pmatrix}\right)$$

...relabeling:

```
vyDistribution = SetLabel[%, {v, y}]
```

$$\begin{pmatrix} v \\ y \end{pmatrix} \sim \mathcal{N}\left(\begin{pmatrix} p_0 \\ (\alpha + v\,\beta)\,(1+\gamma) - v\,(\beta + \beta\,\gamma) + (\beta + \beta\,\gamma)\,p_0 \end{pmatrix}, \begin{pmatrix} \Sigma_0 & (\beta + \beta\,\gamma)\,\Sigma_0 \\ (\beta + \beta\,\gamma)\,\Sigma_0 & (\beta + \beta\,\gamma)^2\,\Sigma_0 + \sigma_u^2 \end{pmatrix}\right)$$

The conditional distribution is:

```
vConditionalDist = MVNConditional[vyDistribution, v, y] // Simplify
```

$$v \sim \mathcal{N}\left(\frac{-\beta\,(1+\gamma)\,(-y + \alpha + \alpha\,\gamma)\,\Sigma_0 + \sigma_u^2\,p_0}{\beta^2\,(1+\gamma)^2\,\Sigma_0 + \sigma_u^2}, \frac{\Sigma_0\,\sigma_u^2}{\beta^2\,(1+\gamma)^2\,\Sigma_0 + \sigma_u^2}\right)$$

Market efficiency:

```
pEqu = GetMean[vConditionalDist] == (p /. pRule)
```

$$\frac{-\beta\,(1+\gamma)\,(-y + \alpha + \alpha\,\gamma)\,\Sigma_0 + \sigma_u^2\,p_0}{\beta^2\,(1+\gamma)^2\,\Sigma_0 + \sigma_u^2} == y\,\lambda + \mu$$

```
pSolutions = Reduce[∀y pEqu, {μ, λ}]
```

$$\beta^2\,\Sigma_0 + 2\,\beta^2\,\gamma\,\Sigma_0 + \beta^2\,\gamma^2\,\Sigma_0 + \sigma_u^2 \neq 0 \;\&\&$$
$$\mu == \frac{-\alpha\,\beta\,\Sigma_0 - 2\,\alpha\,\beta\,\gamma\,\Sigma_0 - \alpha\,\beta\,\gamma^2\,\Sigma_0 + \sigma_u^2\,p_0}{\beta^2\,\Sigma_0 + 2\,\beta^2\,\gamma\,\Sigma_0 + \beta^2\,\gamma^2\,\Sigma_0 + \sigma_u^2} \;\&\&\; \lambda == \frac{\beta\,\Sigma_0 + \beta\,\gamma\,\Sigma_0}{\beta^2\,\Sigma_0 + 2\,\beta^2\,\gamma\,\Sigma_0 + \beta^2\,\gamma^2\,\Sigma_0 + \sigma_u^2}$$

```
pSolutions = Simplify @ ToRules[Take[pSolutions, -2]]
```

$$\left\{\mu \to \frac{-\alpha\,\beta\,(1+\gamma)^2\,\Sigma_0 + \sigma_u^2\,p_0}{\beta^2\,(1+\gamma)^2\,\Sigma_0 + \sigma_u^2},\ \lambda \to \frac{\beta\,(1+\gamma)\,\Sigma_0}{\beta^2\,(1+\gamma)^2\,\Sigma_0 + \sigma_u^2}\right\}$$

```
EquationSet = Apply[Equal, Join[pSolutions, xSolutions], 1]
```

$$\left\{\mu == \frac{-\alpha\,\beta\,(1+\gamma)^2\,\Sigma_0 + \sigma_u^2\,p_0}{\beta^2\,(1+\gamma)^2\,\Sigma_0 + \sigma_u^2},\ \lambda == \frac{\beta\,(1+\gamma)\,\Sigma_0}{\beta^2\,(1+\gamma)^2\,\Sigma_0 + \sigma_u^2},\ \alpha == -\frac{\mu}{2\,\lambda + 2\,\gamma\,\lambda},\ \beta == \frac{1}{2\,\lambda + 2\,\gamma\,\lambda}\right\}$$

```
ModelSolutions = Simplify[Solve[EquationSet, {μ, λ, α, β}], {Σ0 > 0, σu^2 > 0}] // First
```

$$\left\{\alpha \to -\frac{\sqrt{\frac{\sigma_u^2}{\Sigma_0}}\ p_0}{1+\gamma},\ \mu \to p_0,\ \lambda \to \frac{1}{2}\sqrt{\frac{\Sigma_0}{\sigma_u^2}},\ \beta \to \frac{\sqrt{\frac{\sigma_u^2}{\Sigma_0}}}{1+\gamma}\right\}$$

To recover the original solution ...

```
ModelSolutions /. γ → 0
```

$$\left\{\alpha \to -\sqrt{\frac{\sigma_u^2}{\Sigma_0}}\ p_0,\ \mu \to p_0,\ \lambda \to \frac{1}{2}\sqrt{\frac{\Sigma_0}{\sigma_u^2}},\ \beta \to \sqrt{\frac{\sigma_u^2}{\Sigma_0}}\right\}$$

The expected profits are

```
Simplify[PowerExpand[Eπ /. xRule /. ModelSolutions]]
```

$$\frac{\sqrt{\sigma_u^2}\ (v - p_0)^2}{2\,(1+\gamma)\,\sqrt{\Sigma_0}}$$

The informed trader's demand is:

```
Simplify[xRule /. ModelSolutions]
```

$$x \to \frac{\sqrt{\frac{\sigma_u^2}{\Sigma_0}}\ (v - p_0)}{1+\gamma}$$

Since $\gamma > 0$, the informed trader's expected profits are lower. Also, she fades her demand to take into account the crooked broker. The informativeness of the price is:

```
GetVariance[vConditionalDist] /. ModelSolutions
```

$$\frac{\Sigma_0}{2}$$

Unchanged, relative to the original model.

The multiperiod model

Setup

There are $k = 1, \ldots, N$ auctions. These are equally-spaced on a unit time interval. In real time, the *k*th auction occurs at time $\frac{k}{T}$, so the increment between auctions is $\Delta t = \frac{1}{T}$. At the *kth* auction, noise traders submit an order flow $u_k \sim N\left(0, \sigma_u^2 \Delta t\right)$. The informed trader submits an order flow Δx_t.

The informed traders profits are given recursively as $\pi_k = (v - p_k)\, \Delta\, x_k + \pi_{k+1}$ for $k = 1,\ \ldots,\ N$ and $\pi_{N+1} \equiv 0$.

Solution

Kyle's Theorem 2 gives the solution as follows
The informed trader's demand in auction *n* is linear in the difference between the true value *v* and the price on the preceding auction, p_{n-1}:

```
ClearAll[α, β, λ, δ, Σ, Δt];
```

```
Notation[βn_ ⟺ β[n_]];
Notation[λn_ ⟺ λ[n_]];
Notation[αn_ ⟺ α[n_]];
Notation[δn_ ⟺ δ[n_]];
Notation[Σn_ ⟺ Σ[n_]];
Notation[ϕn_ ⟺ ϕ[n_]];
```

```
ΔxSolution = Δxn_ :→ βn (v - pn-1) Δt;
```

The MM's price adjustment rule is linear in the total order flow:

```
ΔpSolution = Δpn_ :→ λn (Δun + Δxn);
```

```
(Δx2 /. ΔxSolution) + u2
```

$u_2 + \Delta t\ (v - p_1)\ \beta_2$

```
Δp1 /. ΔpSolution
```

$(\Delta u_1 + \Delta x_1)\ \lambda_1$

Expected profits are quadratic:

```
EπSolution = Eπn_ :→ αn-1 (v - pn-1)^2 + δn;
```

The constants in the above are given by the solutions to the difference equation system:

```
αSolution = α[k_] :→ 1/(4 λ[k+1] (1 - α[k+1] λ[k+1]));
δSolution = δ[k_] :→ Δt α[k+1] λ[k+1]^2 σu^2 + δ[k+1];
βSolution = β[n_] :→ (1 - 2 α[n] λ[n])/(Δt (2 λ[n] (1 - α[n] λ[n])));
λSolution = λ[n_] :→ β[n] Σ[n]/σu^2;
```

$$\alpha\text{Solution} = \alpha_{k\_} :\to \frac{1}{4\,\lambda_{k+1}\,(1-\alpha_{k+1}\,\lambda_{k+1})};$$
$$\delta\text{Solution} = \delta_{k\_} :\to \Delta t\,\alpha_{k+1}\,\lambda_{k+1}^2\,\sigma_u^2 + \delta_{k+1};$$
$$\beta\text{Solution} = \beta_{n\_} :\to \frac{1-2\,\alpha_n\,\lambda_n}{\Delta t\,(2\,\lambda_n\,(1-\alpha_n\,\lambda_n))};$$
$$\lambda\text{Solution} = \lambda_{n\_} :\to \frac{\beta_n\,\Sigma_n}{\sigma_u^2};$$

$$\text{TerminalConditions}[n\_] := \{\alpha_n \to 0,\ \delta_n \to 0\}$$

subject to the terminal conditions $\alpha_N = \delta_N = 0$. The above recursions are backwards. Σ_n is the variance of v conditional on all order flow and prices through auction n. It is given by the *forward* recursion:

$$\Sigma\text{Solution} = \Sigma_{n\_} :\to (1-\beta_n\,\lambda_n\,\Delta t)\,\Sigma_{n-1};$$

The solutions for $\{\alpha_k, \delta_k, \beta_k, \lambda_k, \Sigma_k\}$ don't depend on the realization of *v*. That is, given $\{\Sigma_0, p_0, \sigma_u^2\}$, agents can perfectly forecast the depth and demand coefficients.

Analysis of solution

To compute a solution given *N* and the model parameters $\{\Sigma_0, p_0, \sigma_u^2\}$, start at the *nth* auction. Taking the solution for λ_n and plugging in from the solution for β_n yields a cubic polynomial equation for λ_n

$$\lambda\text{Equation} = \lambda_n == (\lambda_n \;/.\; \lambda\text{Solution} \;/.\; \beta\text{Solution})$$

$$\lambda_n == \frac{(1-2\,\alpha_n\,\lambda_n)\,\Sigma_n}{2\,\Delta t\,\sigma_u^2\,\lambda_n\,(1-\alpha_n\,\lambda_n)}$$

The equation has three roots. They are not pretty ones. If you really want to see them, run the following *Mathematica* line (which is not visible in the pdf/printout versions of this document).

```
(*s=Simplify[Solve[λEquation,λn],{σu^2>0,Σn>0,αn>0,Δt>0}];*)
```

The full solution procedure is as follows. The model parameters are Σ_0, v and σ_u^2.

Pick a trial value of Σ_N. By the terminal conditions, $\alpha_N = \delta_N = 0$. Solve the polynomial equation for λ_N. In general, this is a cubic, but at step N, it is quadratic. Take λ_N as the positive root. Compute β_n using βSolution. Compute Σ_{N-1} using:

$$\Sigma\text{SolutionBack} = \Sigma_{n\_} :> -\frac{\Sigma_{n+1}}{-1+\Delta t\,\beta_{n+1}\,\lambda_{n+1}}$$

$$\Sigma_{n\_} :\to -\frac{\Sigma_{n+1}}{-1+\Delta t\,\beta_{n+1}\,\lambda_{n+1}}$$

At step $N-1$, compute α_{N-1} and δ_{N-1} using αSolution and δSolution. Solve for λ_{N-1}, taking the middle root. Compute β_{n-1} using βSolution. Compute Σ_{n-1} using ΣSolutionBack. Iterate over

$N-2,\ N-3,\ \ldots 1$. Denote the sequence of Σ_k implied by the intial guess Σ_N as $\Sigma^*_{N-1},\ \Sigma^*_{N-2},\ \ldots\ \Sigma^*_0$. Compare Σ^*_0 to the actual value Σ_0. Make a new guess at Σ_N. Repeat until Σ^*_0 has (sufficiently) converged to Σ_0.

KyleSolve$[T,\ \sigma^2_u,\ \Sigma_0]$ computes a full numerical solution, where T is the number of auctions and σ^2_u is the total noise variance (assumed equally distributed over all the auctions). *Mathematica* reserves "N" to denote the numerical rounding operator, so these routines use *T* instead.

□ *Multiple times >2*

The routine KyleSolve$[T,\ \sigma^2_u,\ \Sigma_0]$ solves the multiperiod model following the procedure described in Kyle (1985, p. 1333).

```
KyleFull[T_Integer, Varu_, Σ0_] := Module[
  {Δt = 1 / T, ϕRoots, λEquation, λRoots, α, δ, λ, Σ, β, ϕ},
  σu^2 = Varu;
  ϕ[T] = 0; Do[
   equ = ϕ[k] - ϕ[k - 1] == -Δt - Δt^2 / ϕ[k - 1] + Δt^3 / ϕ[k - 1]^2;
   ϕRoots = NSolve[equ, ϕ[k - 1]];
   s = ϕ_(k-1) /. ϕRoots;
   s2 = -5 / 4 < (ϕ_k - #) / Δt <= -1 & /@ s;
   s3 = Select[ϕ[k - 1] /. ϕRoots, -5 / 4 < (ϕ[k] - #) / Δt <= -1 &];
   ϕ[k - 1] = First[s3];,
   {k, T, 1, -1}];
  Σ[0] = Σ0;
  Do[Σ[k] = Σ[k - 1] / (1 + Δt / ϕ[k - 1]);
   α[k] = Sqrt[σu^2 ϕ[k] / (4 Σ[k])] ;
   λEquation = λ[k] == ((1 - 2 α[k] λ[k]) Σ[k]) / (2 Δt σu^2 λ[k] (1 - α[k] λ[k]));
   λRoots = Chop [ λ[k] /. NSolve[λEquation, λ[k]]];
   λ[k] =
    If[Length[λRoots] ≤ 2, First[Select[λRoots, # > 0 &]], Sort[λRoots][[2]]];
   β[k] = λ[k] σu^2 / Σ[k];,
   {k, T}];
  δ[T] = 0; Do[δ[k] = Δt α[k + 1] λ[k + 1]^2 σu^2 + δ[k + 1], {k, T - 1, 1, -1}];
  Table[{α[k], δ[k], λ[k], β[k], Σ[k], Σ[k - 1]}, {k, 1, T}]
 ];
KyleFull[2, 1, 12]
```

```
{{0.122689, 0, 2.26165, 0.272348, 8.30426, 12},
 {0, 0, 2.03768, 0.490755, 4.15213, 8.30426}}
```

```
KyleHeadings = TableHeadings → {Automatic, {"α", "δ", "λ", "β", "Σk", "Σk-1"}};
```

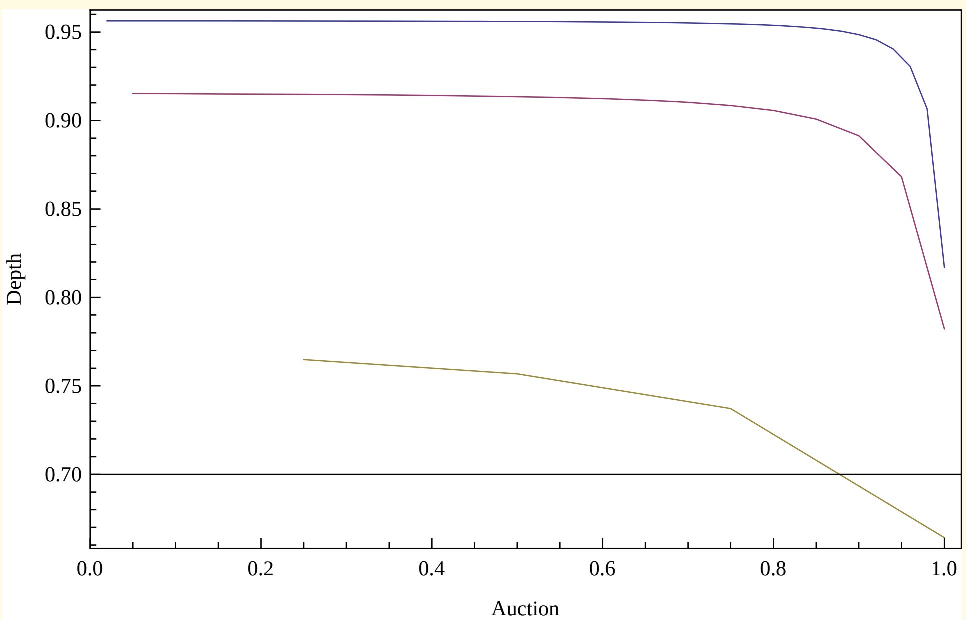
```
TableForm[KyleFull[10, 1, 12.], KyleHeadings]
```

	α	δ	λ	β	Σ_k	Σ_{k-1}
1	0.151997	0.948289	2.99755	0.271968	11.0217	12.
2	0.150532	0.813222	2.99544	0.298459	10.0364	11.0217
3	0.148713	0.680041	2.99258	0.330952	9.04235	10.0364
4	0.146389	0.549293	2.98856	0.371825	8.03754	9.04235
5	0.143305	0.421807	2.98264	0.42495	7.01881	8.03754
6	0.138994	0.298924	2.97336	0.497102	5.98138	7.01881
7	0.132489	0.183043	2.95745	0.601412	4.91751	5.98138
8	0.121375	0.0791026	2.92636	0.767444	3.81313	4.91751
9	0.0973351	0	2.85076	1.08034	2.63877	3.81313
10	0	0	2.56845	1.9467	1.31938	2.63877

```
Needs["ErrorBarPlots`"]
```

```
λBuild[T_, vu_, Σ0_] := Module[{Δt = 1 / T, λ},
   λ = Transpose[KyleFull[T, vu, Σ0]][[3]];
   Transpose[{Range[Δt, 1, Δt], λ}]];
```

```
ListPlot[{λBuild[50, 1, 1.`], λBuild[20, 1, 1.`], λBuild[4, 1, 1.`]},
 Joined → True, PlotRange → {{0, 1.02`}, All}, Background → GrayLevel[1],
 Frame → True, FrameTicks → {Automatic, Automatic, None, None},
 FrameLabel → {"Auction", "Depth", None, None}, BaseStyle → {FontFamily → "Times"}]
```



Problem based on Holden and Subrahmanyam (not referenced in book).

There are two identical informed traders `i = 1, 2`. Each conjectures that the MM uses a linear price adjustment rule.

```
pRule = p → y λ + μ;
```

where y is the total order flow:

```
yRule = y → u + x1 + x2;
```

The first (representative) informed trader's profits are:

```
πRule = π1 → (v - p) x1;
```

Substituting in for the price conjecture and y:

```
π1 /. πRule /. pRule /. yRule
```

$$x_1\ (v - \mu - \lambda\ (u + x_1 + x_2))$$

Informed trader 1 conjectures that trader 2 uses a linear strategy:

```
xRule = x2 → α + β v;
```

The expected profits are:

```
ℰπ1 = π1 /. πRule /. pRule /. yRule /. xRule /. u → 0
```

$$x_1\ (v - \mu - \lambda\ (\alpha + v\ \beta + x_1))$$

Solve the model.

Analysis

The informed trader maximizes expected profits by trading:

```
xOpt = (Take @@ Flatten @@ Solve[∂x1 ℰπ1 == 0, x1] // Simplify)
```

$$x_1 \to -\frac{\alpha\ \lambda + v\ (-1 + \beta\ \lambda) + \mu}{2\ \lambda}$$

The market maker's problem

The MM conjectures that each informed trader's demand is linear in `v`:

```
xRule = x_ → α + v β;
```

Knowing the optimization process that the informed traders followed, the MM can solve for α and β:

```
xEqu = (x1 /. xRule) == (x1 /. xOpt)
```

$$\alpha + v\,\beta == -\frac{\alpha\,\lambda + v\,(-1 + \beta\,\lambda) + \mu}{2\,\lambda}$$

for all v :

```
r = Reduce[ForAll[v, True, xEqu], {α, β}, Reals]
```

$$\left(\lambda < 0\ \&\&\ \alpha == -\frac{\mu}{3\,\lambda}\ \&\&\ \beta == \frac{1}{3\,\lambda}\right)\ ||\ \left(\lambda > 0\ \&\&\ \alpha == -\frac{\mu}{3\,\lambda}\ \&\&\ \beta == \frac{1}{3\,\lambda}\right)$$

```
xSolutions = ToRules[r[[2, {2, 3}]]]
```

$$\left\{\alpha \to -\frac{\mu}{3\,\lambda},\ \beta \to \frac{1}{3\,\lambda}\right\}$$

Now the MM must figure out $E[V \mid Y = y]$. First, consider the joint distribution of v and u:

```
uvDist = MVN[{{p0}, {0}}, {{Σ0, 0}, {0, σu^2}}, {v, u}]
```

$$\begin{pmatrix} v \\ u \end{pmatrix} \sim \mathcal{N}\left(\begin{pmatrix} p_0 \\ 0 \end{pmatrix}, \begin{pmatrix} \Sigma_0 & 0 \\ 0 & \sigma_u^2 \end{pmatrix}\right)$$

Now consider the joint distribution of v and y where y =

```
y /. yRule /. xRule
```

$$u + 2\,\alpha + 2\,v\,\beta$$

So

```
MakeLinearForm[uvDist, {v, y /. yRule /. xRule}]
```

$$\begin{pmatrix} v \\ u + 2\,\alpha + 2\,v\,\beta \end{pmatrix} \sim \mathcal{N}\left(\begin{pmatrix} p_0 \\ 2\,\alpha + 2\,\beta\,p_0 \end{pmatrix}, \begin{pmatrix} \Sigma_0 & 2\,\beta\,\Sigma_0 \\ 2\,\beta\,\Sigma_0 & 4\,\beta^2\,\Sigma_0 + \sigma_u^2 \end{pmatrix}\right)$$

Relabeling:

```
vyDist = SetLabel[%, {v, y}]
```

$$\begin{pmatrix} v \\ y \end{pmatrix} \sim \mathcal{N}\left(\begin{pmatrix} p_0 \\ 2\,\alpha + 2\,\beta\,p_0 \end{pmatrix}, \begin{pmatrix} \Sigma_0 & 2\,\beta\,\Sigma_0 \\ 2\,\beta\,\Sigma_0 & 4\,\beta^2\,\Sigma_0 + \sigma_u^2 \end{pmatrix}\right)$$

Now consider the conditional distribution $v \mid y$:

```
vConditionalDist = MVNConditional[vyDist, v, y]
```

$$v \sim \mathcal{N}\left(p_0 + \frac{2\,\beta\,\Sigma_0\,(y - 2\,\alpha - 2\,\beta\,p_0)}{4\,\beta^2\,\Sigma_0 + \sigma_u^2},\ \Sigma_0 - \frac{4\,\beta^2\,\Sigma_0^2}{4\,\beta^2\,\Sigma_0 + \sigma_u^2}\right)$$

Market efficiency requires that the mean of this distribution be equal to p.

```
GetMean[vConditionalDist] == (p /. pRule)
```

$$p_0 + \frac{2\,\beta\,\Sigma_0\,(y - 2\,\alpha - 2\,\beta\,p_0)}{4\,\beta^2\,\Sigma_0 + \sigma_u^2} == y\,\lambda + \mu$$

for all values of y:

```
r = Reduce[ForAll[y, True, %], {μ, λ}]
```

$$4\,\beta^2\,\Sigma_0 + \sigma_u^2 \neq 0\ \&\&\ \mu == \frac{-4\,\alpha\,\beta\,\Sigma_0 + \sigma_u^2\,p_0}{4\,\beta^2\,\Sigma_0 + \sigma_u^2}\ \&\&\ \lambda == \frac{2\,\beta\,\Sigma_0}{4\,\beta^2\,\Sigma_0 + \sigma_u^2}$$

```
pSolutions = ToRules[r[[{2, 3}]]]
```

$$\left\{\mu \to \frac{-4\,\alpha\,\beta\,\Sigma_0 + \sigma_u^2\,p_0}{4\,\beta^2\,\Sigma_0 + \sigma_u^2},\ \lambda \to \frac{2\,\beta\,\Sigma_0}{4\,\beta^2\,\Sigma_0 + \sigma_u^2}\right\}$$

Collecting these results...

```
EquationSet = (Equal @@ #1 &) /@ Join[pSolutions, xSolutions]
```

$$\left\{\mu == \frac{-4\,\alpha\,\beta\,\Sigma_0 + \sigma_u^2\,p_0}{4\,\beta^2\,\Sigma_0 + \sigma_u^2},\ \lambda == \frac{2\,\beta\,\Sigma_0}{4\,\beta^2\,\Sigma_0 + \sigma_u^2},\ \alpha == -\frac{\mu}{3\,\lambda},\ \beta == \frac{1}{3\,\lambda}\right\}$$

... and solving for the parameters in terms of the fundamental variables.

```
ModelSolutions = Solve[EquationSet, {μ, λ, α, β}]; ModelSolutions // TableForm
```

$$\begin{array}{llll}
\alpha \to -\frac{\sqrt{\sigma_u^2}\,p_0}{\sqrt{2}\,\sqrt{\Sigma_0}} & \mu \to p_0 & \lambda \to \frac{\sqrt{2}\,\sqrt{\Sigma_0}}{3\,\sqrt{\sigma_u^2}} & \beta \to \frac{\sqrt{\sigma_u^2}}{\sqrt{2}\,\sqrt{\Sigma_0}} \\
\alpha \to \frac{\sqrt{\sigma_u^2}\,p_0}{\sqrt{2}\,\sqrt{\Sigma_0}} & \mu \to p_0 & \lambda \to -\frac{\sqrt{2}\,\sqrt{\Sigma_0}}{3\,\sqrt{\sigma_u^2}} & \beta \to -\frac{\sqrt{\sigma_u^2}}{\sqrt{2}\,\sqrt{\Sigma_0}}
\end{array}$$

Only first solution has positive λ and β, so:

```
ModelSolutions = First @ ModelSolutions; TableForm @ ModelSolutions
```

$$\alpha \to -\frac{\sqrt{\sigma_u^2}\,p_0}{\sqrt{2}\,\sqrt{\Sigma_0}}$$

$$\mu \to p_0$$

$$\lambda \to \frac{\sqrt{2}\,\sqrt{\Sigma_0}}{3\,\sqrt{\sigma_u^2}}$$

$$\beta \to \frac{\sqrt{\sigma_u^2}}{\sqrt{2}\,\sqrt{\Sigma_0}}$$

Properties of the solution

An informed trader's expected profits are:

```
Simplify @ PowerExpand @ (ℰπ₁ /. x_Rule //. Model_Solutions)
```

$$\frac{\sqrt{\sigma_u^2}\,(v-p_0)^2}{3\sqrt{2}\,\sqrt{\Sigma_0}}$$

Either (say, the first) informed trader's demand is

```
x₁ /. x_Rule /. Model_Solutions // Simplify
```

$$\frac{\sqrt{\sigma_u^2}\,(v-p_0)}{\sqrt{2}\,\sqrt{\Sigma_0}}$$

How much of the private information is impounded in the price?

```
GetVariance[vConditionalDist] // Simplify
```

$$\frac{\Sigma_0\,\sigma_u^2}{4\,\beta^2\,\Sigma_0+\sigma_u^2}$$

Or, in terms of the input parameters:

```
Simplify[% //. Model_Solutions]
```

$$\frac{\Sigma_0}{3}$$