SOME NOTES ON VAR MODELS FOR THE EZ.

LAFC

Define a set of endogenous variables $Y_t = [y_{1t}, ..., y_{k_Y t}]$, $t = 1, ..., T$, and define a set of exogenous variables X_t stored in vectors of dimension $1 \times k_Y$ and $1 \times k_X$, respectively. Note that Y_t could be the first difference of some variables, I use this notation for the sake of generality. Let D_t be the vector of deterministic components of the model. D_t could contain an intercept, a linear trend, or dummies. Also define $Z_t' = [Y_{t-1}', ..., Y_{t-p}', X_t', D_t']$. The main model we consider is a VARX(p) of the form

$$Y_t = D_t^* + \sum_{l=1}^{p} Y_{t-l} A_l^* + X_t B^* + \epsilon_t,$$

$$= D_t^* + Z_t \Gamma^* + \epsilon_t^*,$$

(1) $$y_{it} = d_{it}^* + Z_t \gamma_i^* + \epsilon_{it}^*,$$

The parameter matrix Γ^* is of dimensions $k_Y \times (pk_Y + k_X(:= k))$. The VAR is estimated equation-by-equation both to speed up computations and to adapt the penalty to the variance in each equation. A possible exception is if using the group Lasso for lag selection.

ESTIMATION

Lasso. The objective function of the Lasso is given by

(2) $$L(\gamma_i) = \frac{1}{T}\|y_i - d_{it} - Z\gamma_i\|^2 + 2\lambda_T \|\gamma_i\|_{\ell_1}.$$

The model is estimated equation-by-equation and the penalty parameter is selected by BIC among those on the path selected by `glmnet`. `Lassovar` takes care of this part.

Adaptive Lasso. The objective function of the adaptive Lasso is given by

(3) $$L(\gamma_i) = \frac{1}{T}\|y_i - d_{it} - Z\gamma_i\|^2 + 2\lambda_T \sum_{j=1}^{k} w_{ij}\gamma_{ij}.$$

where the adaptive weights can be given by $w_{ij} = \frac{1}{\hat{\gamma}_{ij}}$ where $\hat{\gamma}_{ij}$ is some initial estimator of γ_{ij}^*. If $\hat{\gamma}_{ij} = 0$, variable j is excluded from equation i before estimating the adaptive Lasso. OLS or the Lasso are fine initial estimators, but they both have weaknesses. The risk of misclassification for the Lasso (excluding a relevant variable) and the poor quality of the estimates if k is close to T.

SHOCK RESPONSE

Consider a VAR(1) in difference where the parameters have been estimated:

$$\Delta Y_t = \Delta Y_{t-1}\hat{A} + \epsilon_t,$$

To compute the response to an exogenous shock of size s to the first variable in Y, construct the vector $\delta = [s, 0, ..., 0]$. The response at after h periods is given by $\delta\hat{A}^h$.

- For models with multiple lags the companion form (or simply recursive computations) can be used.

Date: June 18, 2015.

- For models with exogenous variables, at path for the exogenous variables has to be specified. In general it should be equal to zero or to a random walk forecast, at least in periods following an initial shock.
- The deterministics can be omitted.
- For VECM models, things are more complicated but not much more difficult.

CONDITIONAL FORECASTS

The options so far:

- Create a model for the exogenous variable.
- Use naïve methods (i.e. RW).
- Compute prediction density for the exogenous variable and plug in the model. Gaussian + linear = Gaussian.
- Condition on true value.

SOME TESTS

```
> library(lassovar)
> library(ggplot2)
> library(reshape2)
> load('../Result/vardata')
> # loadind and subseting the data
> dat <- data.frame(vardataframe[116:180, c(1,2,6,11,13,17,8,24) ])
> # differentiating
> difdat <- tail(dat,-1) - head(dat,-1)
>
> # estimate a VAR(1)
> lv_lev<-lassovar(dat,lags=1, ic="BIC")
> summary(lv_lev)
Call:
lassovar(dat = dat, lags = 1, ic = "BIC")


Model estimated equation by equation
Selection criterion: BIC
Estimator: Lasso
Dimensions: T = 64  N = 8


Total number of variables selected:26 (40.625% of candidates)
[1] "Model summary statistics:"
          Lambda non-zero     resid var
CPE    0.009137410        3 1.679413e-04
DJES   0.047738176        3 7.856802e+01
HICP   0.002310290        4 2.891006e-04
LTN    0.006771786        3 6.916725e-02
M3     0.014432675        3 1.286696e-03
POILU  0.037676144        4 2.943369e-01
LHO    0.018900280        1 1.011330e-03
YED    0.007223302        5 1.002183e-04
```

```
> lv_lev_trend<-lassovar(dat,lags=1, ic="BIC",exo=seq(1:nrow(dat)))
> summary(lv_lev_trend)

Call:
lassovar(dat = dat, exo = seq(1:nrow(dat)), lags = 1, ic = "BIC")


Model estimated equation by equation
Selection criterion: BIC
Estimator: Lasso
Dimensions: T = 64  N = 8


Total number of variables selected:30 (41.66667% of candidates)
[1] "Model summary statistics:"
          Lambda non-zero    resid var
CPE    0.009137410        4 1.667344e-04
DJES   0.052392590        4 7.243895e+01
HICP   0.002310290        5 2.821701e-04
LTN    0.006771786        3 6.916725e-02
M3     0.014432675        3 1.286696e-03
POILU  0.009332668        5 2.867100e-01
LHO    0.018900280        1 1.011330e-03
YED    0.007223302        5 1.002183e-04

> lv_dif<-lassovar(difdat,lags=1, ic="BIC")
> summary(lv_dif)

Call:
lassovar(dat = difdat, lags = 1, ic = "BIC")


Model estimated equation by equation
Selection criterion: BIC
Estimator: Lasso
Dimensions: T = 63  N = 8


Total number of variables selected:16 (25% of candidates)
[1] "Model summary statistics:"
           Lambda non-zero    resid var
CPE    0.0024551858        0 8.912461e-05
DJES   1.8958393270        2 8.264576e+01
HICP   0.0014289972        4 2.665823e-04
LTN    0.0778170470        0 8.896670e-02
M3     0.0009786965        6 8.803950e-04
POILU  0.1861160585        0 3.695596e-01
LHO    0.0029187219        2 2.666321e-04
YED    0.0014462239        2 4.939361e-05

>

> # printing the diagonal of the parameter matrix
> ar1 <- cbind(diag(coef(lv_lev)[-1,]),
+              diag(coef(lv_lev_trend)[2:(1+ncol(dat)),]),
+              diag(coef(lv_dif)[-1,]))
```

```
> rownames(ar1) <- lv_lev$var.names
> colnames(ar1) <- c('level','level + trend','diff')
> mar1 <- melt(ar1)
> ggplot(mar1,aes(x=Var1,y=value,colour=Var2,shape=Var2)) +
+   geom_point(size=3) + theme_minimal()
```

The parameters fo the VAR in level are pretty close to 1. The trend is usually set to zero by the Lasso. It should probably not be penalized anyway. In the model in difference, the estimator often does not select an equation's own lag in most cases. weird.

```
> rownames(ar1) <- lv_lev$var.names
> colnames(ar1) <- c('level','level + trend','diff')
> mar1 <- melt(ar1)
> ggplot(mar1,aes(x=Var1,y=value,colour=Var2,shape=Var2)) +
+   geom_point(size=3) + theme_minimal()
```