Compare forecast rules

2026 January 13

```
source("simple-fcrules.R")
source("esm-fcrules.R")
source("linear-fcrules.R")
# The formats of forecast functions are the following.
# esm_fc = function(train,holdout,alpha,level,iprint) # simple exp smoothing
# lholt_fc = function(train,holdout,alpha,beta,level,slope,iprint) # Holt linear
# Winters multipliocate seasonal
# mseason_fc = function(train,holdout,alpha,beta,gamma,level,slope,season,iprint)
# aseason_fc = function(train,holdout,alpha,beta,gamma,level,slope,season,iprint)

# Some of these functions are to be coded in the labs and submitted on canvas.
```

Compare forecast rules for Vancouver monthly total precipitation

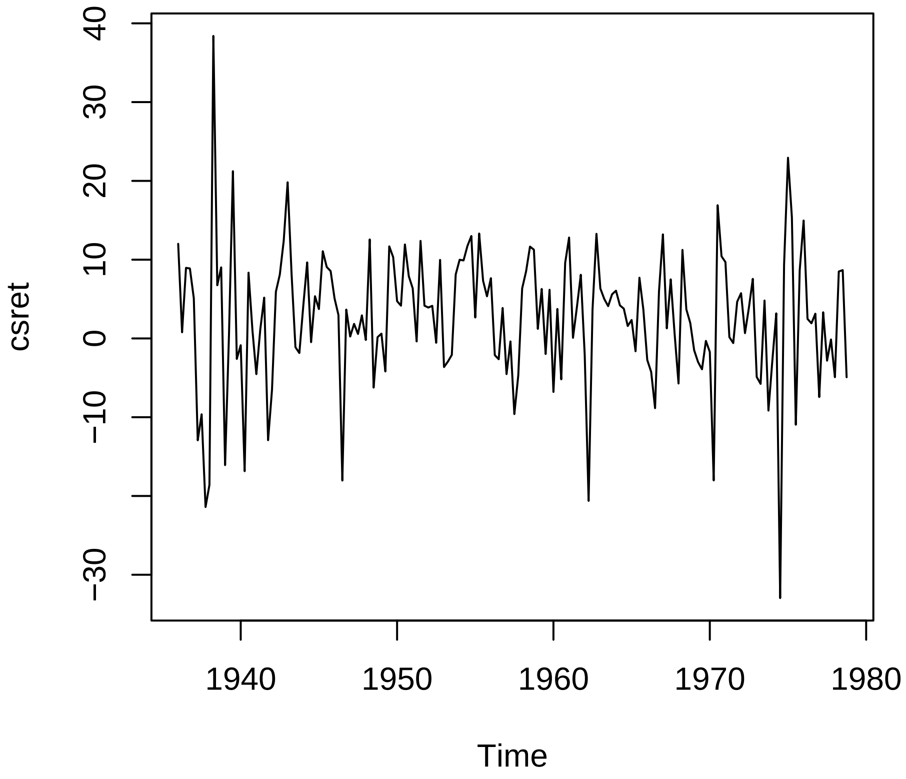
```
v = read.csv("vanc-prec-temp.csv",header=T)
print(names(v))
#> [1] "yearmon"   "totprecip" "meantemp"
# length is 86*12 = 1032
print(nrow(v))
#> [1] 1032
nn = nrow(v)
summary(v$totprecip)
#>    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
#>    0.00   42.40   80.20   94.14  134.22  350.80

izero = which(v$totprecip==0)
print(length(izero))
#> [1] 5

# Change 0s to 1, to apply Winters multiplicative rule
v$totprecip[izero] = 1

# holdout set: last 18 years
ntrain = 12*68
vtrain = v$totprecip[1:ntrain]
vholdout = v$totprecip[(ntrain+1):nn]
mon_holdout = v$yearmon[(ntrain+1):nn]

z = ts(vtrain,start=c(1938,1),frequency=12)

# Winters additive seasonal
wafit = HoltWinters(z,seasonal="additive")
# trend column means estimated slope
```

```
print(wafit$fitted[(ntrain-24):(ntrain-12),])
#>             xhat    level         trend     season
#>  [1,] 168.65468 111.4245 -0.0125203412  57.242670
#>  [2,] 160.42644 112.5719 -0.0008883238  47.855468
#>  [3,]  99.92680 117.8627  0.0521815766 -17.988057
#>  [4,] 111.91683 114.7029  0.0199690713  -2.806023
#>  [5,]  85.18426 115.9621  0.0323972819 -30.810235
#>  [6,]  68.45548 116.2921  0.0353823039 -47.872038
#>  [7,]  51.77469 116.3361  0.0354683131 -64.596871
#>  [8,]  41.39168 116.2425  0.0341740895 -74.885009
#>  [9,]  41.90104 116.4077  0.0354883302 -74.542180
#> [10,]  60.80161 115.6539  0.0275724776 -54.879873
#> [11,] 130.48609 115.2541  0.0232865787  15.208671
#> [12,] 177.02359 116.7558  0.0381136059  60.229629
#> [13,] 173.76923 114.3952  0.0140562968  59.360013


print(wafit)
#> Holt-Winters exponential smoothing with trend and additive seasonal component.
#> 
#> Call:
#> HoltWinters(x = z, seasonal = "additive")
#> 
#> Smoothing parameters:
#>  alpha: 0.05934165
#>  beta : 0.01002888
#>  gamma: 0.1151639
#> 
#> Coefficients:
#>              [,1]
#> a   113.639601105
#> b     0.006337913
#> s1   57.515631040
#> s2  -23.851607411
#> s3   -0.543751456
#> s4  -30.266880342
#> s5  -47.856381561
#> s6  -64.832455292
#> s7  -74.645781286
#> s8  -75.983080581
#> s9  -55.660022806
#> s10  17.907592476
#> s11  55.850545092
#> s12  57.955057930

# $fitted is missing for first year
names(wafit)
#> [1] "fitted"       "x"            "alpha"        "beta"         "gamma"
#> [6] "coefficients" "seasonal"     "SSE"          "call"

print(sqrt(wafit$SSE/(ntrain-12)))
#> [1] 46.55144
```

```
# Winters multiplicative seasonal
wmfit = HoltWinters(z,seasonal="multiplicative")
# trend column means estimated slope
print(wmfit$fitted[(ntrain-24):(ntrain-12),])
#>             xhat    level     trend     season
#>  [1,] 166.27251 359.6965 0.2484171 0.46193874
#>  [2,] 166.41004 360.6346 0.2610454 0.46110296
#>  [3,]  82.77851 363.5168 0.3090421 0.22752237
#>  [4,] 112.89577 361.4646 0.2658043 0.31209924
#>  [5,]  77.43871 362.6569 0.2827707 0.21336524
#>  [6,]  64.03205 363.8086 0.2986822 0.17586038
#>  [7,]  38.76947 364.4847 0.3055924 0.10627879
#>  [8,]  33.50187 366.2708 0.3327031 0.09138449
#>  [9,]  40.61840 368.2089 0.3621004 0.11020507
#> [10,]  76.50405 366.9866 0.3330880 0.20827645
#> [11,] 142.92315 365.7220 0.3038323 0.39047281
#> [12,] 176.94594 366.4901 0.3123330 0.48240121
#> [13,] 174.78376 365.5873 0.2900830 0.47771124

alpha = wafit$alpha;  beta = wafit$beta; gamma = wafit$gamma
level = wafit$coef[1];  slope = wafit$coef[2]; season = wafit$coefficient[3:14]
aseason = aseason_fc(vtrain,vholdout,alpha,beta,gamma,level,slope,season,iprint=F)

alph = wmfit$alpha;  bet = wmfit$beta; gamm = wmfit$gamma
leve = wmfit$coef[1];  slop = wmfit$coef[2]; seaso = wmfit$coefficient[3:14]
mseason = mseason_fc(vtrain,vholdout,alph,bet,gamm,leve,slop,seaso,iprint=F)

persbm = persistbymonth_fc(vtrain,vholdout,iprint=F)
iidbm = iidbymonth_fc(vtrain,vholdout,iprint=F)

out = cbind(mon_holdout/100,vholdout, aseason$fc, mseason$fc, persbm$fc, iidbm$fc)
colnames(out) = c("yearmon","holdout","add_seasonal","mult_seasonal","persist_mon","iid_bymon")
print(round(out[1:12,],2))
#>       yearmon holdout add_seasonal mult_seasonal persist_mon iid_bymon
#>  [1,] 2006.01   283.6       171.16        190.35       249.6    151.10
#>  [2,] 2006.02    57.0        96.54         74.10        45.8    114.73
#>  [3,] 2006.03    92.4       117.55        120.01       132.8    103.88
#>  [4,] 2006.04    70.0        86.37         81.60        90.2     72.55
#>  [5,] 2006.05    42.8        67.83         65.61        68.6     57.44
#>  [6,] 2006.06    54.4        49.38         41.56        49.6     47.67
#>  [7,] 2006.07    25.2        39.88         36.11        43.6     33.96
#>  [8,] 2006.08     4.8        37.68         37.15        28.6     37.37
#>  [9,] 2006.09    39.4        56.03         69.28        53.6     57.45
#> [10,] 2006.10    57.8       128.59        143.20       155.4    119.04
#> [11,] 2006.11   350.8       162.26        161.34       136.6    158.98
#> [12,] 2006.12   146.0       175.60        169.24       160.8    172.26

rmse_vec = c(aseason$rmse, mseason$rmse, persbm$rmse, iidbm$rmse)

cat(round(rmse_vec,2), "\n")
#> 45.29 47.55 64.13 44.05

# Interpret: how do the methods compare?
```

Compare forecast rules for quarterly corporate stock returns

```
f = read.csv("CSGBCBRL36to78.csv", skip=3, header=T)

csret = f$CS # returns of corporate stocks
csret = ts(csret, start=c(1936,1), end=c(1978,4), frequency=4)
plot(csret)
```

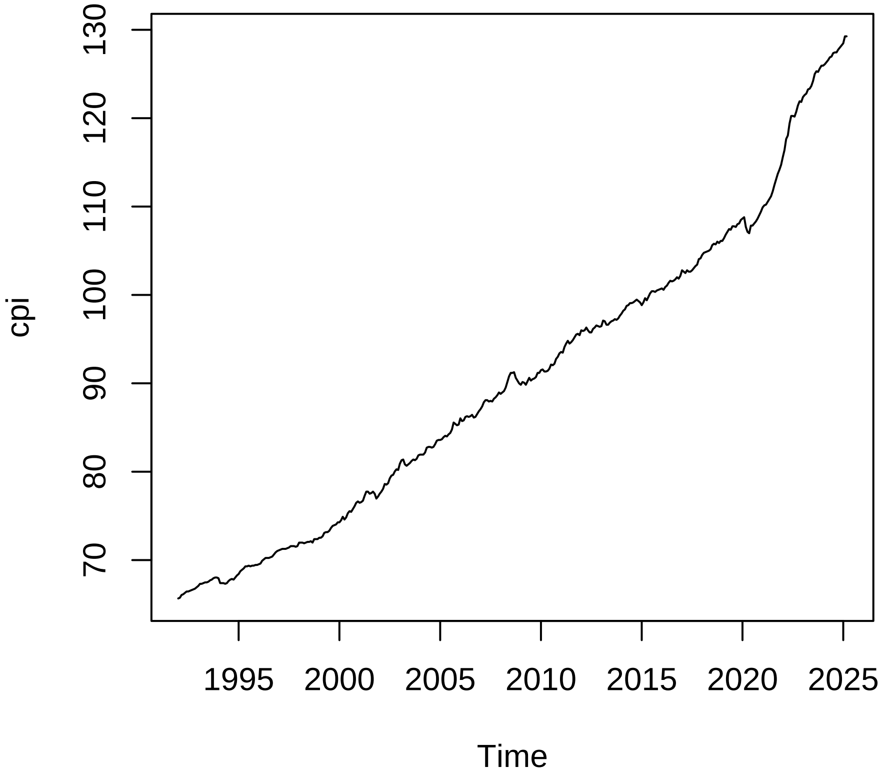


```
nn = length(csret) # 172 quarters = 43 years

# training set 35 years, ntrain = 140
ntrain = 35*4
train = csret[1:ntrain]
holdout = csret[(ntrain+1):nn]
qu_holdout = f$quarter[(ntrain+1):nn]

fit_expsmo = HoltWinters(train, beta=F, gamma=F)
names(fit_expsmo)
#> [1] "fitted"       "x"            "alpha"        "beta"         "gamma"
#> [6] "coefficients" "seasonal"     "SSE"          "call"
print(fit_expsmo)
#> Holt-Winters exponential smoothing without trend and without seasonal component.
#>
```

```
#> Call:
#> HoltWinters(x = train, beta = F, gamma = F)
#>
#> Smoothing parameters:
#>  alpha: 0.09226752
#>  beta : FALSE
#>  gamma: FALSE
#>
#> Coefficients:
#>       [,1]
#> a 1.729562

#esmfc = esm_fc(train,holdout, alpha=fit_expsmo$alpha,
#  level=fit_expsmo$fitted[ntrain-1,2], iprint=F)
# below is the correction
esmfc = esm_fc(train,holdout, alpha=fit_expsmo$alpha,
  level=fit_expsmo$coefficients[1], iprint=F)

fit_hw = HoltWinters(train, gamma=F)
#holtfc = lholt_fc(train,holdout, alpha=fit_hw$alpha, beta=fit_hw$beta,
#  level=fit_hw$fitted[ntrain-2,2],
#  slope=fit_hw$fitted[ntrain-2,3], iprint=F)
# below is the correction
holtfc = lholt_fc(train,holdout, alpha=fit_hw$alpha, beta=fit_hw$beta,
  level=fit_hw$coefficients[1],
  slope=fit_hw$coefficients[2], iprint=F)

persistfc = persist_fc(train,holdout,iprint=F)
iidfc = iid_fc(train,holdout,iprint=F)

out2 = cbind(qu_holdout/100,holdout, esmfc$fc, holtfc$fc, persistfc$fc, iidfc$fc)
colnames(out2) = c("quarter","holdout","expsmo","holt","persist","iid")
print(round(out2[1:12,],2))
#>       quarter holdout expsmo  holt persist  iid
#>  [1,]   71.01    9.69   1.73  6.90   10.43 2.87
#>  [2,]   71.02    0.17   2.46  9.63    9.69 2.87
#>  [3,]   71.03   -0.59   2.25  6.13    0.17 2.87
#>  [4,]   71.04    4.66   1.99  3.29   -0.59 2.87
#>  [5,]   72.01    5.74   2.24  4.17    4.66 2.87
#>  [6,]   72.02    0.67   2.56  5.26    5.74 2.87
#>  [7,]   72.03    3.91   2.39  3.24    0.67 2.87
#>  [8,]   72.04    7.56   2.53  3.61    3.91 2.87
#>  [9,]   73.01   -4.89   2.99  5.77    7.56 2.87
#> [10,]   73.02   -5.77   2.26  0.54   -4.89 2.87
#> [11,]   73.03    4.81   1.52 -3.27   -5.77 2.87
#> [12,]   73.04   -9.16   1.83 -0.01    4.81 2.87

rmse_vec = c(esmfc$rmse, holtfc$rmse, persistfc$rmse, iidfc$rmse)

cat(round(rmse_vec,2),"\n")
#> 10.03 11.99 13.68 9.7

# Interpret: how do the methods compare?
```

Compare forecast rules for monthly CPI in Canada

```
data = read.csv("CANCPALTT01IXOBSAM.csv",header=T)
cpi = ts(data$CANCPALTT01IXOBSAM,start=c(1992,1), end=c(2025,3),frequency=12)
plot(cpi)
```



```
nn = length(cpi)  # monthly consumer price index, Canada
print(nn)
#> [1] 399

ntrain = round(nn*0.8)
train = cpi[1:ntrain]
holdout = cpi[(ntrain+1):nn]
ymd_holdout = data$observation_date[(ntrain+1):nn]
library(lubridate)
mon_holdout = 100* year(ymd_holdout) + month(ymd_holdout)

cpi_hw = HoltWinters(train, gamma=F)
#holtfc = lholt_fc(train, holdout, alpha=cpi_hw$alpha, beta=cpi_hw$beta,
#  level=cpi_hw$fitted[ntrain-2,2],
#  slope=cpi_hw$fitted[ntrain-2,3], iprint=F)
holtfc = lholt_fc(train, holdout, alpha=cpi_hw$alpha, beta=cpi_hw$beta,
  level=cpi_hw$coefficients[1],
  slope=cpi_hw$coefficients[2], iprint=F)

persistfc = persist_fc(train,holdout,iprint=F)
iidfc = iid_fc(train,holdout,iprint=F)

reg = lm(train[-1] ~ train[-ntrain])
summary(reg)
#>
#> Call:
#> lm(formula = train[-1] ~ train[-ntrain])
#>
#> Residuals:
#>      Min       1Q   Median       3Q      Max
#> -0.76598 -0.11196 -0.02661  0.12313  0.66359
#>
#> Coefficients:
#>                Estimate Std. Error t value Pr(>|t|)
#> (Intercept)    0.029712   0.088043   0.337    0.736
#> train[-ntrain] 1.001142   0.001037 965.821   <2e-16 ***
#> ---
#> Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
#>
#> Residual standard error: 0.221 on 316 degrees of freedom
#> Multiple R-squared:  0.9997, Adjusted R-squared:  0.9997
#> F-statistic: 9.328e+05 on 1 and 316 DF,  p-value: < 2.2e-16

bcoef = reg$coeff
print(bcoef)
#>    (Intercept) train[-ntrain]
#>     0.02971201     1.00114171

linearfc = linear_fc(train,holdout,bcoef,iprint=F)

out3 = cbind(mon_holdout/100,holdout, holtfc$fc, linearfc$fc, persistfc$fc, iidfc$fc)
colnames(out3) = c("yearmon","holdout","holt","linear","persist","iid")
print(round(out3[1:12,],2))
#>       yearmon holdout   holt linear persist   iid
#>  [1,] 2018.08  105.79 105.77 105.79  105.64 84.16
#>  [2,] 2018.09  105.72 105.93 105.94  105.79 84.16
#>  [3,] 2018.10  106.03 105.85 105.87  105.72 84.16
#>  [4,] 2018.11  105.87 106.17 106.18  106.03 84.16
#>  [5,] 2018.12  106.11 106.01 106.02  105.87 84.16
#>  [6,] 2019.01  106.11 106.24 106.26  106.11 84.16
#>  [7,] 2019.02  106.43 106.24 106.26  106.11 84.16
#>  [8,] 2019.03  106.82 106.56 106.58  106.43 84.16
#>  [9,] 2019.04  107.14 106.96 106.97  106.82 84.16
#> [10,] 2019.05  107.45 107.28 107.29  107.14 84.16
#> [11,] 2019.06  107.37 107.59 107.61  107.45 84.16
#> [12,] 2019.07  107.77 107.51 107.53  107.37 84.16

rmse_vec = c(holtfc$rmse, linearfc$rmse, persistfc$rmse, iidfc$rmse)
cat(round(rmse_vec,3),"\n")
#> 0.365 0.373 0.458 33.053
```

```
# Interpret: how do the methods compare?
```