

ADVANCED PROGRAMMING IN R

PROJECT: INTERACTIVE MODELS WITH R

Ismayil Ismayilov | Katarzyna Piotrowska

HOW TO RUN THE APP

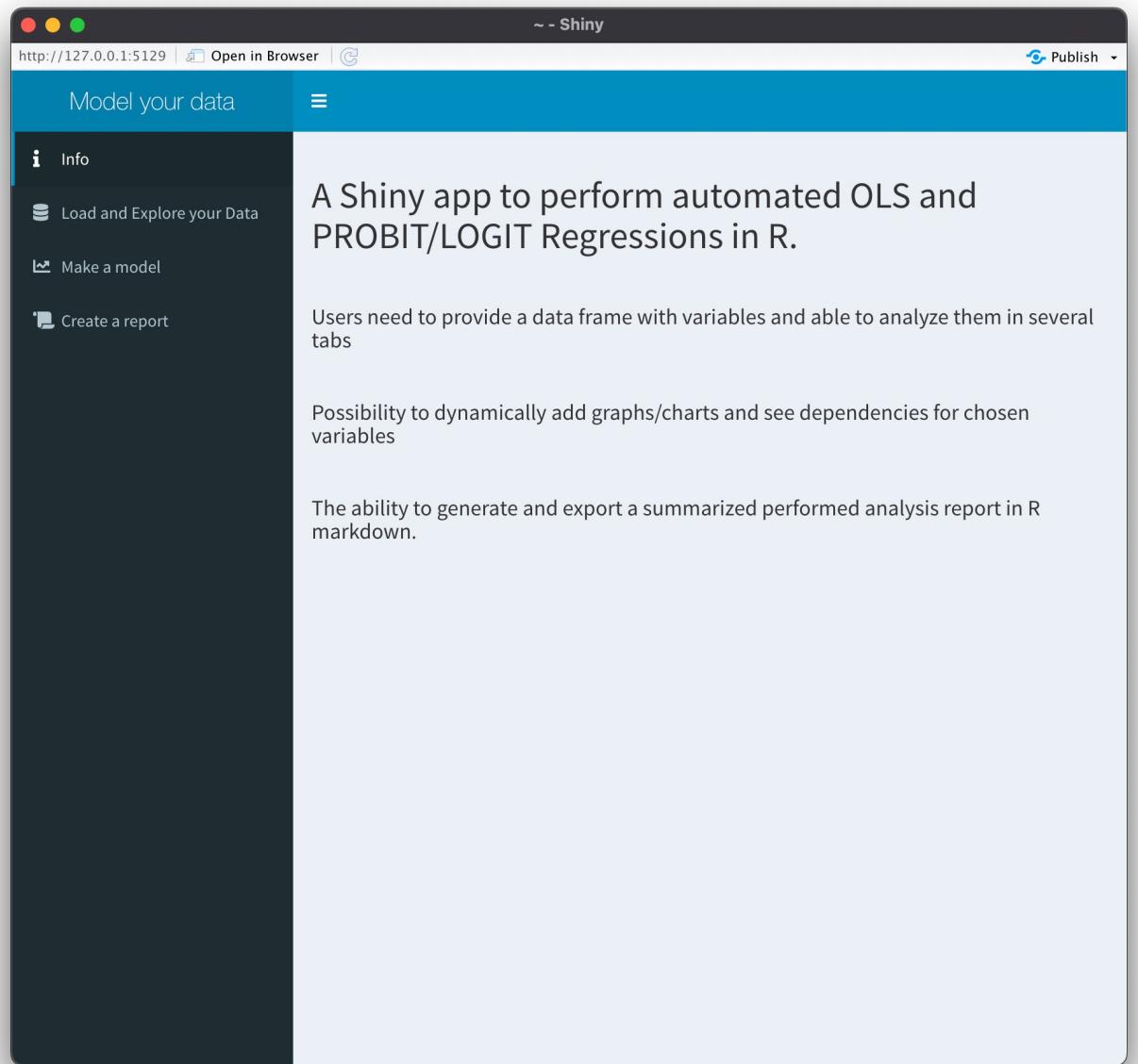
In order to run the app successfully, please **follow the instructions** below:

1. Put all the files in one folder:
 - a. File names: App.r; functions.R; Report.Rmd; Server.R; ui.R
2. Open the App.R file with R Studio and set the direction to the same folder above.
3. Run all the codes.
4. If R Studio shows an errors run the codes **line by line** (*starting from line 15*) without quitting or removing any environment data.
 - a. Example errors:
 - i. Error in hasGroups(choices) : object 'sep' not found
 - ii. Error in force(ui) : object 'ui' not found

NOTE: In order to check the app, you may use "testing_file.csv" file which included among the all other files.

DESCRIPTION & MAIN PAGE

A Shiny app to perform automated OLS and PROBIT/LOGIT Regressions in R.



EXPLORING DATA

- Users needs to provide a data frame with variables.
 - “Header” for variables
 - Separator & Decimal Points options

The screenshot shows a user interface for exploring data. On the left, a sidebar menu includes 'Info', 'Load and Explore your Data' (which is currently selected), 'Make a model', and 'Create a report'. The main area is titled 'Load the data' and shows a file upload section where 'costofliving.csv' has been uploaded. It includes options for 'Header' (unchecked), 'Separator' (set to 'comma'), and 'Decimal points' (set to 'dot'). Below these are 'Load' and 'Delete missing values' buttons. To the right, a 'Select sub set' panel allows users to filter data by 'Rank' (with dropdowns for 'Rank' and 'Change Column Name') and provides a 'Delete rows' button. At the bottom, a table displays data for two cities:

Rank	City	Cost.of.Living.Index	Rent.Index	Cost.of.Living.Plus.Rent.Index	Groceries.Index	Restaurant.Price.Index	Local.Purchasing.Power.Index
	Hamilton, Bermuda	149.02	96.10	124.22	157.89	155.22	79.43
	Zurich, Switzerland	131.24	69.26	102.19	136.14	132.52	129.79

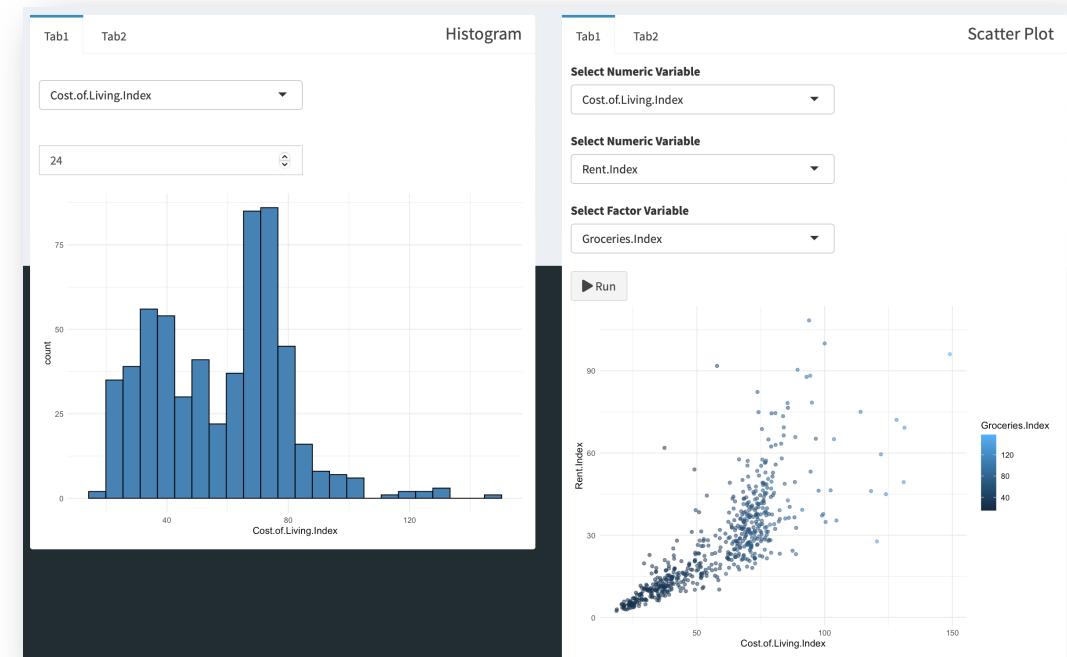
EXPLORING DATA

Show 5 entries

Rank	City	Cost.of.Living.Index	Rent.Index	Cost.of.Living.Plus.Rent.Index	Groceries.Index
	Hamilton, Bermuda	149.02	96.10	124.22	157.89
	Zurich, Switzerland	131.24	69.26	102.19	136.14
	Basel, Switzerland	130.93	49.38	92.70	137.07
	Zug, Switzerland	128.13	72.12	101.87	132.61
	Lugano, Switzerland	123.99	44.99	86.96	129.17

Showing 1 to 5 of 578 entries

Previous 1 2 3 4 5 ... 116 Next



The user is able to dynamically add graphs/charts and see dependencies for the chosen variables.

REGRESSION

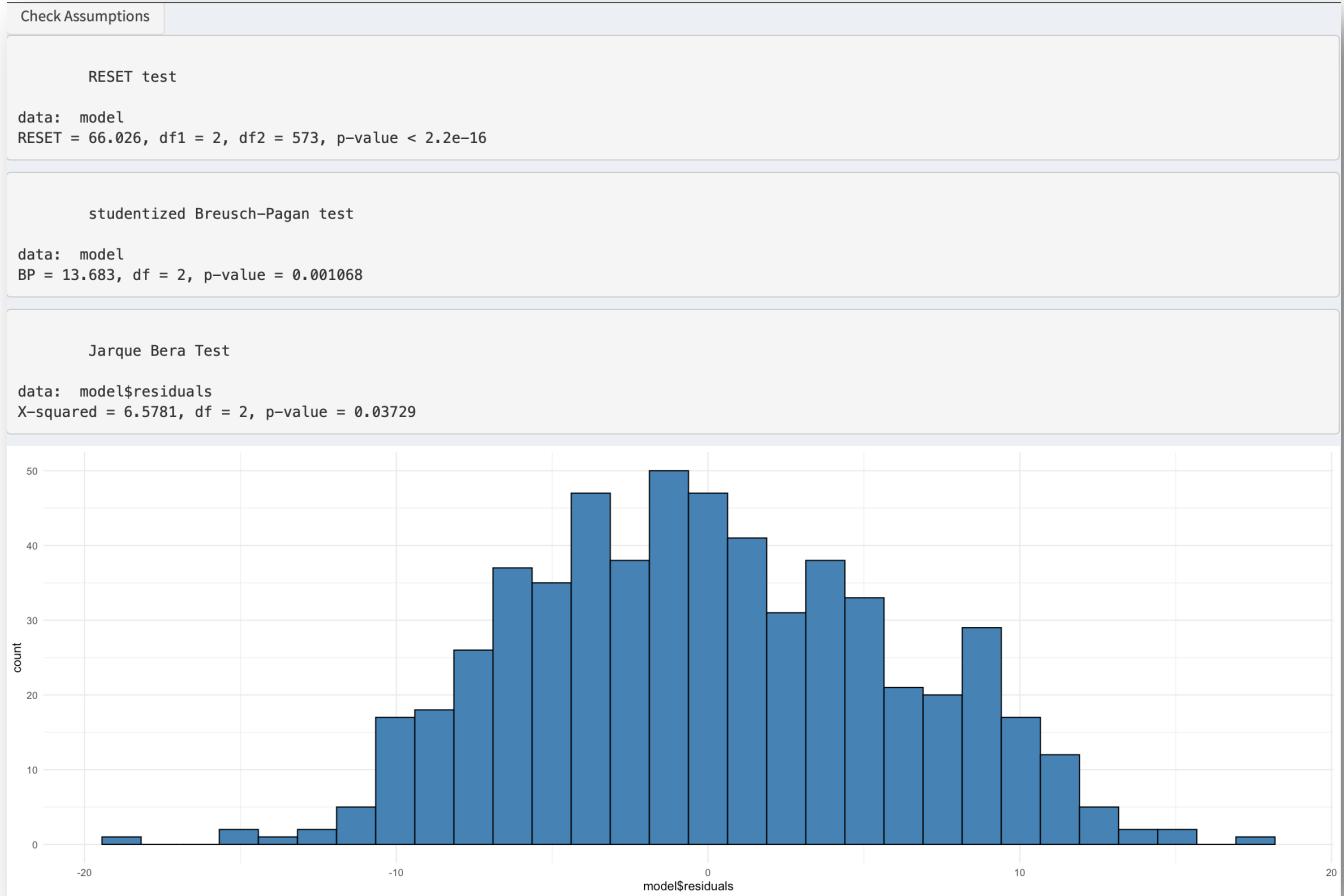
- Choose OLS and PROBIT/LOGIT Regressions
- As simple as Drag and Drop
 - User needs to drag the variables from the left table to the right side. Noting that the first variable will be dependents, and all others dependent.

The screenshot shows a user interface for regression analysis. At the top, there are two tabs: "OLS" (selected) and "PROBIT/LOGIT". Below the tabs, a section titled "Select variables to your model" contains two columns: "Drag from here" and "to here". The "Drag from here" column lists variables: Rank, City, Rent.Index, Cost.of.Living.Plus.Rent.Index, and Restaurant.Price.Index. The "to here" column lists variables: Cost.of.Living.Index, Groceries.Index, and Local.Purchasing.Power.Index. Below this, a large box labeled "▶ Estimate" displays the R command used to run the regression, the resulting output including coefficients, standard errors, t-values, and p-values, and the residual statistics.

```
Call:  
lm(formula = formula, data = data())  
  
Residuals:  
    Min      1Q  Median      3Q     Max  
-18.5040 -4.3443 -0.3835  4.3228 17.8685  
  
Coefficients:  
              Estimate Std. Error t value Pr(>|t|)  
(Intercept) 6.411882  0.673972  9.514 < 2e-16 ***  
Groceries.Index 0.882539  0.015254 57.856 < 2e-16 ***  
Local.Purchasing.Power.Index 0.053909  0.009866  5.464 6.95e-08 ***  
---  
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1  
  
Residual standard error: 6.011 on 575 degrees of freedom  
Multiple R-squared:  0.9232,   Adjusted R-squared:  0.923  
F-statistic: 3458 on 2 and 575 DF,  p-value: < 2.2e-16
```

OLS | ASSUMPTION

- Possibility to check the OLS regression model assumptions
 - RAMSEY RESET TEST
 - Breusch-Pagan Test
 - Jarque Bera Test
 - Normality/ Distribution



Probit & Logit | Marginal effects and Assumptions

- In Probit and Logit models, it is important to interpret the marginal results. In the app, a user can click on the "Calculate marginal effect" to achieve the results.
- Possibility to check the Probit & Logit regression model assumptions:
 - Test of joint insignificance of all variables in the model
 - Linktest - to check if a specification of the model is correct

The screenshot shows a Shiny application window with the URL `http://127.0.0.1:3367`. The interface includes input fields for 'Restaurant.Price.Index' and 'Local.Purchasing.Power.Index'. Below these, a text area displays the R code for the model:

```
Rank.Level ~ Cost.of.Living.Index  
<environment: 0x7fe89c9e0630>
```

A '▶ Estimate' button is present. The main output area contains the following R output:

```
Call:  
glm(formula = formula, family = binomial(link = "probit"), data = data())  
  
Deviance Residuals:  
Min 10 Median 30 Max  
-0.004574 0.000000 0.000000 0.000000 0.003881  
  
Coefficients:  
Estimate Std. Error z value Pr(>|z|)  
(Intercept) -2262.86 17595.97 -0.129 0.898  
Cost.of.Living.Index 22.59 175.67 0.129 0.898  
  
(Dispersion parameter for binomial family taken to be 1)  
  
Null deviance: 1.2437e+02 on 577 degrees of freedom  
Residual deviance: 3.5977e-05 on 576 degrees of freedom  
AIC: 4  
  
Number of Fisher Scoring iterations: 25
```

At the bottom, there is a text input field for comments and a 'Calculate marginal effect' button.

The screenshot shows a Shiny application window with the URL `http://127.0.0.1:3367`. The interface includes a 'Calculate marginal effect' button and a text area describing the marginal effect calculation:

Marginal effect for the observation with the average values of each variables in the model's formula

Call:
`probitfx(formula = form_probit, data = data(), atmean = TRUE)`

Marginal Effects:
`dF/dx Std. Err. z P>|z|`

Cost.of.Living.Index	0	0 NaN	NaN
----------------------	---	-------	-----

Check Assumptions

Test of joint insignificance of all variables in the model

The second power of yhat should be insignificant to say that the specification of the model is fine

Likelihood ratio test

Model 1: Rank.Level ~ Cost.of.Living.Index
Model 2: Rank.Level ~ 1
#df LogLik Df Chisq Pr(>Chisq)
1 2 0.000
2 1 -62.183 -1 124.37 < 2.2e-16 ***
Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Linktest - to check if a specification of the model is correct

Call:
`glm(formula = y ~ yhat + yhat2, family = binomial(link = model$family$link))`

Deviance Residuals:
`Min 10 Median 30 Max`

-1.892e-05	-1.615e-06	-1.615e-06	-1.615e-06	1.890e-05
------------	------------	------------	------------	-----------

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

(Intercept)	-1.52997	5983.49027	0.000	1.000
yhat	0.53559	431.91524	0.001	0.999
yhat2	0.01665	14.38379	0.001	0.999

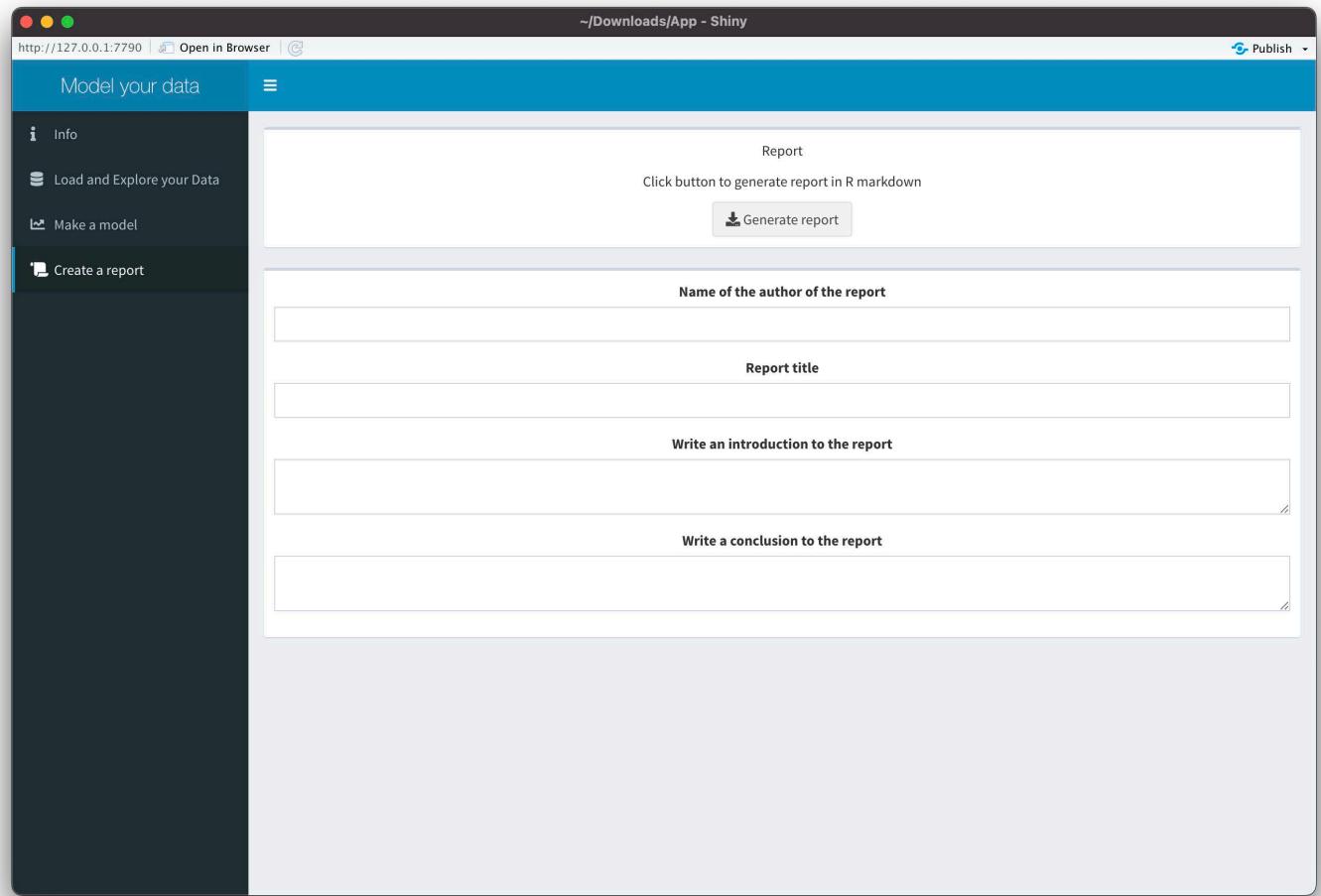
(Dispersion parameter for binomial family taken to be 1)

R Markdown REPORT

A user can generate and export an R Markdown Report.
Before the export, he can add various details to the report
in advance such as:

- Name of the author of the report
- Report Title
- Introduction
- Conclusion

In such all filled, a user will get a fully prepared R
Markdown report to present.



ADVANCED PROGRAMMING ELEMENTS

1. Shiny + creating analytical dashboards
2. Automation of scripts and reports (RMarkdown)
3. Advanced data processing with dplyr, dtplyr, tidyr
4. Own functions in R (including defensive programming)