Analyzing Direct Marketing Data with R

Liang Wei

Brendan Kitts

Lucid Commerce Inc.

Outline

- ## Who we (Lucid Commerce) are
  - Business problems
- ## Why R + SQL Server + SQL Server Reporting Service(SSRS)
  - Analyze data
  - Create and publish analytics product
- ## Case study:
  - Create a boxplot and publish the tool
  - Create a regression model using R and deploy it
- ## Summary

About Lucid Commerce Inc.

- **About Lucid Commerce**
  - Direct response television (DRTV)
    - "advertising that asks consumers to respond directly to the company --- usually either by calling an 800 number or by visiting a web site"
      - -- *http://en.wikipedia.org/wiki/Direct_response_television*
    - *One example of DR Product: credit card application*
  - Business problems
    - Lucid's DRTV optimization
      - Customer targeting, station targeting, TV program targeting
    - Cross channel attribution
      - TV, radio and web.
      - How we can attribute web purchases to an ad on TV
- **Data**
  - Lucid provides data driven business decisions
  - About 3 million direct marketing purchasers as our panel
  - About 400 demographics variables appended to customers
 - Purchase behaviors

Why We Choose R

- Lucid uses Microsoft SQL Server for data management
  - One of the most popular database engines
  - SQL server lacks powerful data visualization
  - SQL Server database engine lacks interactive support for powerful exploration of data using advanced analytics
- R works very well with SQL Server
  - RODBC
- R has richer support for advanced analytics (that we need)
  - Clustering, classification, forecast
- R is powerful in generating graphics
  - Reporting & data visualization
- Quick to publish analytics results and implement R scripts to real-time products

Proceed

R Analytics Workflow
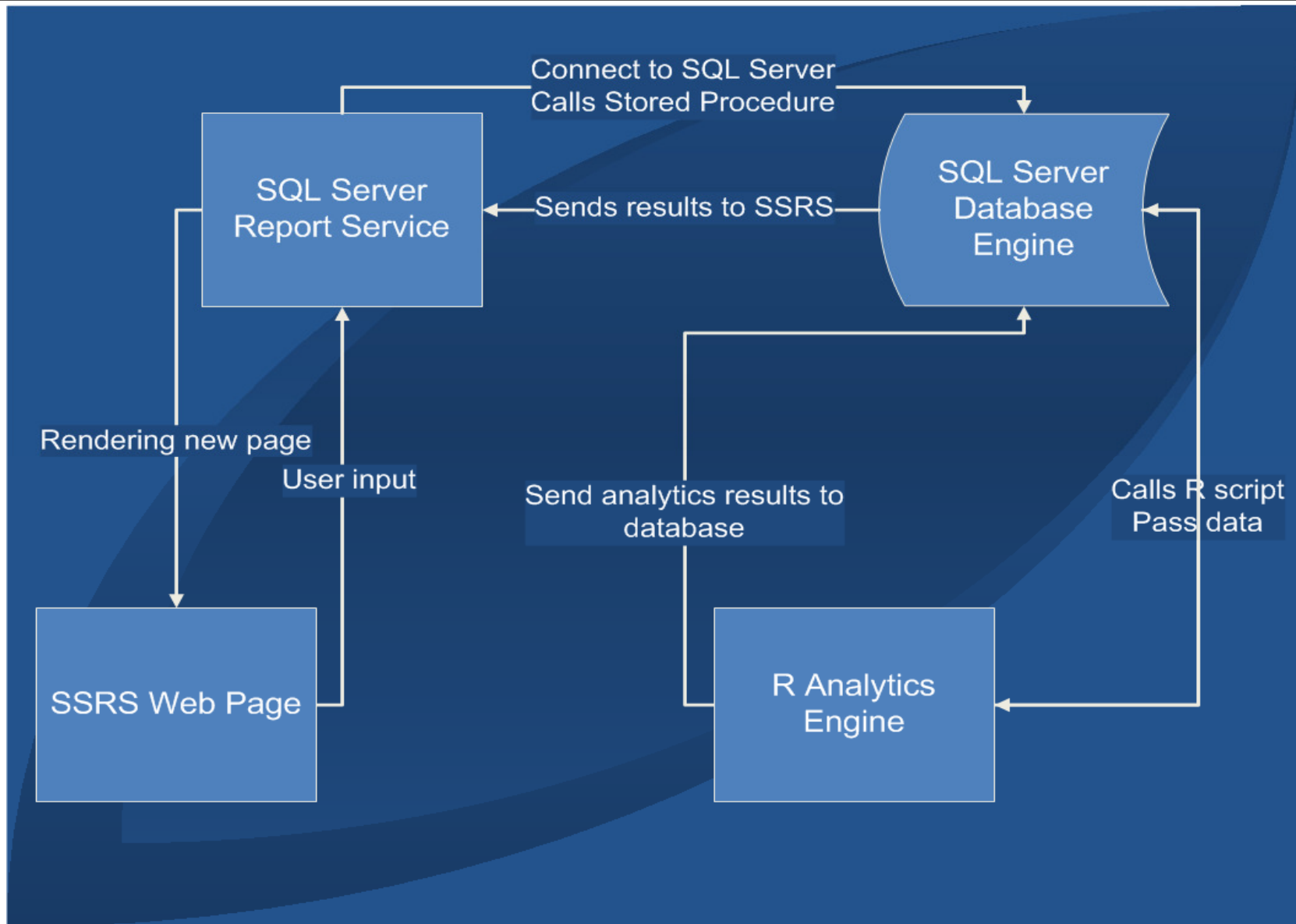
Case Study

- Media buyer's questions
  - How is my campaign performing?
  - What is the best station to place an ad for my client ?
  - What is the best time?
    - Weekday or Weekend
    - Morning or night
  - Local broadcast or national?
  - Visualize variations on different factors
   - boxplot

Case Study – Data Visualization

- ## Example data

	ProjectKey	SpotNaturalKey	RunDateTime	Daypart	DayOfWeek	Weekpart	Station	Channel	Revenue	MER	Orders	MC	SR	DR	WR	CR
1	10021	100210103	02/01/2010 00:30	11pm - 3am	2	Weekday	WMBC	Broadcast	170.84	1.658	2	175.344481619976	208.768395139459	661.995857075067	1882.64504489418	823.224417637082
2	10021	100210434	02/01/2010 07:00	7am - 11am	2	Weekday	DOC	Cable	119.94	0.823117647	1	165.605387081315	287.589431272836	2252.54461156232	1886.37997573535	3622.32923419161
3	10021	100210138	02/01/2010 19:00	6pm - 9pm	2	Weekday	WSAH	Broadcast	170.84	1.3264	2	212.512545225805	287.016600470112	1076.7968221642	1867.0764015486	830.508339561811
4	10021	100210081	02/01/2010 22:30	9pm - 11pm	2	Weekday	WGCB	Broadcast	632.52	3.923869281	4	186.229127099337	338.241140287932	442.548234585608	1882.83712395803	841.841296849395
5	10021	100210435	02/02/2010 08:00	7am - 11am	3	Weekday	DOC	Cable	290.78	2.205437908	3	193.793978604953	283.004401745792	2253.76992161541	1904.33636092042	3621.98769030066
6	10021	100210105	02/02/2010 15:00	3pm - 6pm	3	Weekday	WMBC	Broadcast	230.78	1.88971123	2	232.484151594661	228.467139888828	838.733173677661	1884.52708177118	835.218518931272
7	10021	100210106	02/03/2010 11:30	11am - 3pm	4	Weekday	WMBC	Broadcast	240.88	1.201540107	2	215.429345718384	233.267001993685	1213.77494049497	1858.5055540891	838.60470991955
8	10021	100210139	02/03/2010 19:00	6pm - 9pm	4	Weekday	WSAH	Broadcast	119.94	0.658494118	1	235.012594198998	281.377288637426	1074.77651995199	1876.57497972138	832.555563821369
9	10021	100210057	02/04/2010 05:00	5am - 7am	5	Weekday	SPK	Cable	3194.9	2.416167421	27	2220.64323521481	5783.42623506925	2806.4635248574	1862.71971317891	3604.88999918685
10	10021	100210216	02/04/2010 05:00	5am - 7am	5	Weekday	FITV	Cable	230.78	1.299176471	2	339.404477728751	502.792358120975	2805.65038020932	1886.64075798066	3615.11634592043
11	10021	100210296	02/04/2010 07:00	7am - 11am	5	Weekday	GSN	Cable	1796.3	2.009485294	15	1366.86315240161	1334.03378835197	2254.40917871822	1876.04970403897	3620.42681873431
12	10021	100210262	02/04/2010 09:00	7am - 11am	5	Weekday	ANGL	Cable	320.72	2.686588235	3	171.718540629191	224.459523555315	2253.8578065092	1901.42449850195	3611.49887218721
13	10021	100210107	02/04/2010 15:30	3pm - 6pm	5	Weekday	WMBC	Broadcast	200.78	1.504684492	2	229.046235145929	237.383940965561	831.762899395207	1882.50813802644	819.383558199139
14	10021	100210140	02/04/2010 21:30	9pm - 11pm	5	Weekday	WSAH	Broadcast	119.94	0.658494118	1	210.044822420768	279.670664671225	436.951044420104	1880.51917973641	828.990014989938
15	10021	100210093	02/05/2010 06:30	5am - 7am	6	Weekday	WLNY	Broadcast	380.76	1.044047059	3	433.914790671123	1092.87601988923	2821.49184926441	1886.750719047	828.783617458361
16	10021	100210210	02/05/2010 06:30	5am - 7am	6	Weekday	COM	Cable	3519.88	1.873013841	29	2893.06827100675	3885.02843819137	2802.61142946085	1880.714363604	3614.2806207795
17	10021	100210047	02/05/2010 08:30	7am - 11am	6	Weekday	GSN	Cable	2010.5	1.228256303	18	2370.21021842048	1338.86384820139	2229.77868025987	1861.92071877102	3611.78179488201
18	10021	100210108	02/05/2010 14:00	11am - 3pm	6	Weekday	WMBC	Broadcast	170.84	1.205818182	2	232.3581291894	218.963326327276	1200.58153260576	1869.0163595005	825.668909898143
19	10021	100210271	02/05/2010 16:00	3pm - 6pm	6	Weekday	TCN	Cable	170.84	0.884266667	2	316.53550061785	313.322170054434	842.406571890978	1871.13012559356	3611.65953394301
20	10021	100210030	02/06/2010 05:00	5am - 7am	7	Weekend	FSSW	Cable	384.6	2.510964706	3	211.679247830735	1099.96009707624	2805.06800731278	1909.23360946768	3623.42979968688
21	10021	100210037	02/06/2010 05:30	5am - 7am	7	Weekend	GMC	Cable	843.2	7.706682353	5	200.032005137216	407.281216467539	2811.49888656686	1901.43842858909	3596.66464528889
22	10021	100210258	02/06/2010 12:30	11am - 3pm	7	Weekend	WLNY	Broadcast	2180.26	2.019071895	20	1536.61725394507	1080.56095014575	1225.42556623174	1889.86524838066	839.099413901456
23	10021	100210263	02/06/2010 13:30	11am - 3pm	7	Weekend	ANGL	Cable	383.62	3.926941176	4	167.457772376628	204.309459404793	1221.39214587117	1876.44355961714	3615.12455669038
24	10021	100210230	02/06/2010 15:00	3pm - 6pm	7	Weekend	USHOP2	Cable	722.44	14.73438337	7	72.4896941215363	260.931219220525	835.563349606596	1885.83972870036	3615.33455587128

Proceed

Case Study – Data Visualization

- ## Building SSRS Report
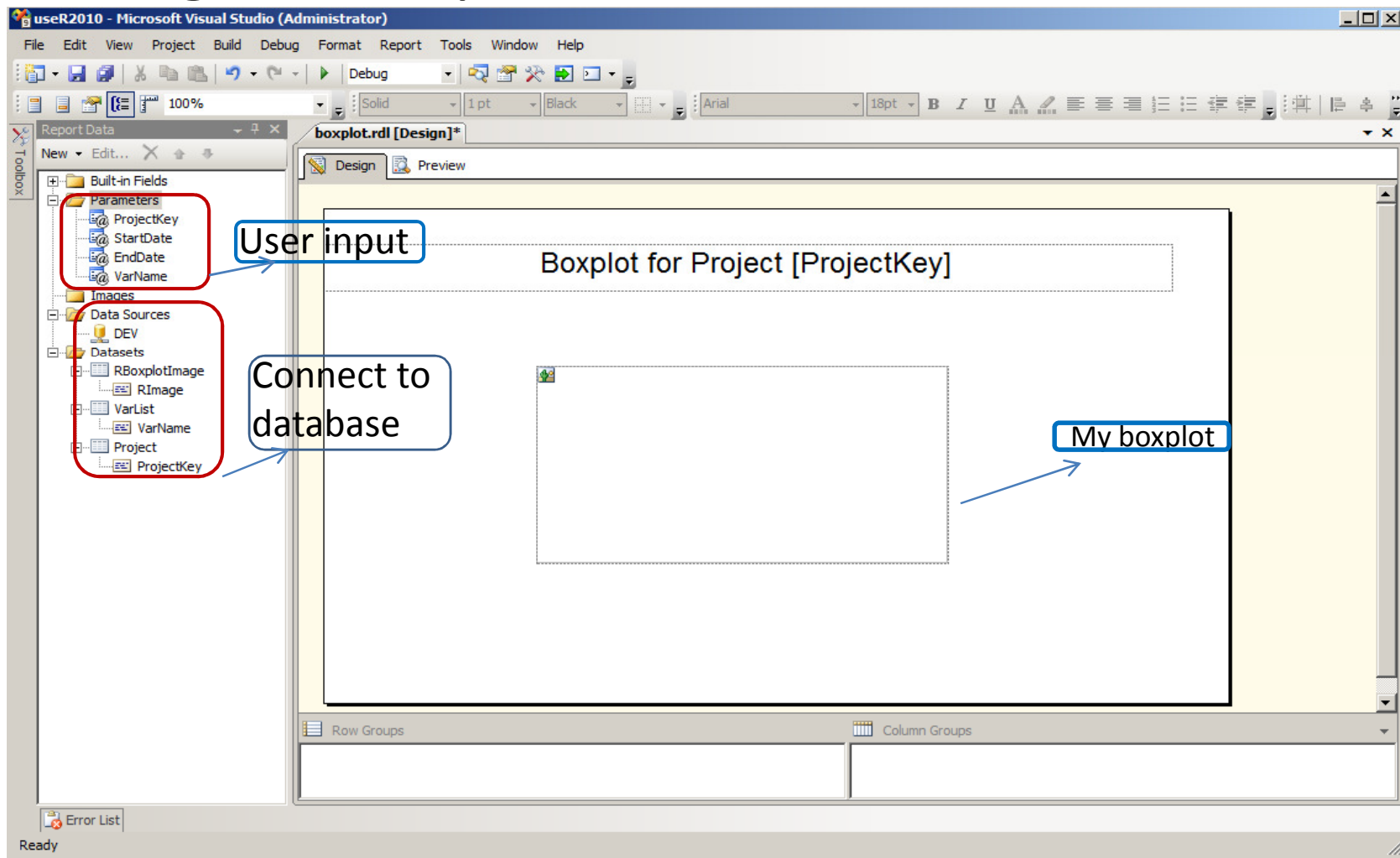
Case Study – Data Visualization

- ## Stored procedure
  - "A stored procedure is a subroutine available to applications accessing a relational database system" *http://en.wikipedia.org/wiki/Stored_procedure*

```sql
1  USE [useR2010]
2  GO
3
4  CREATE PROCEDURE [dbo].[BoxplotProc]
5      @ProjectKey INT,
6      @StartDate DATETIME,
7      @EndDate DATETIME,
8      @VarName VARCHAR(20)
9  AS
10 BEGIN
11     SET NOCOUNT ON;
12     DECLARE @cmd varchar(200)
13     DECLARE @RImageID BIGINT
14     SELECT @RImageID = ABS( CHECKSUM( 1000000000*RAND() ) )
15
16     SET @cmd = 'R --no-save --args projectkey='+ CAST(@ProjectKey   AS varchar(20)) +
17             ' rimageid=' + CAST( @RImageID AS VARCHAR(20)) + ' startdate=''' +
18             CAST( CAST(@StartDate AS DATE) AS VARCHAR(20))  + ''' enddate='''+
19             CAST(CAST(@EndDate AS DATE) AS VARCHAR(20)) +
20             ''' varName=''' +  @VarName + ''' <c:\r\useR2010\boxplot.r'
21
       EXEC master..xp_cmdshell @cmd, NO_OUTPUT

       SELECT RImage FROM useR2010.dbo.boxplotresult where RImageID =  @RImageID

   END
```

Compiles a command string

R --no-save --args projectkey=10021 rimageid=584674235 startdate='2010-01-01' enddate='2010-07-11' varName='revenue' <c:\r\useR2010\boxplot.r

Execute R script from SQL Server with **xp_cmdshell** operating-system command shell

Proceed

- ## R script snippet

```
query <- sprintf(
  "INSERT INTO user2010.dbo.boxplotresult (RImageID, CreatedDateTime, RImage)
  SELECT %d, GETDATE() as CreatedDateTime, *
  FROM  OPENROWSET(BULK N'%s', SINGLE_BLOB) AS import",
  RImageID, file.name
)
sqlQuery(odbc.connect, query)
```

This R code inserts the image we created with R (file.name is the image file's location on my hard disk) to the database. We use SQL Server's bulk load function to load the image file to the database as VARBINARY type filed(which is defined in the table creation step). *

* The authors would like to thank Don Nebres for his advice on this image loading process.

Case study – Data Visualization

- Evaluate the arguments sent from command shell
- Creates graph
- Save results to database
  - *sqlSave:* tables, numbers, texts
  - Bulk load function for images
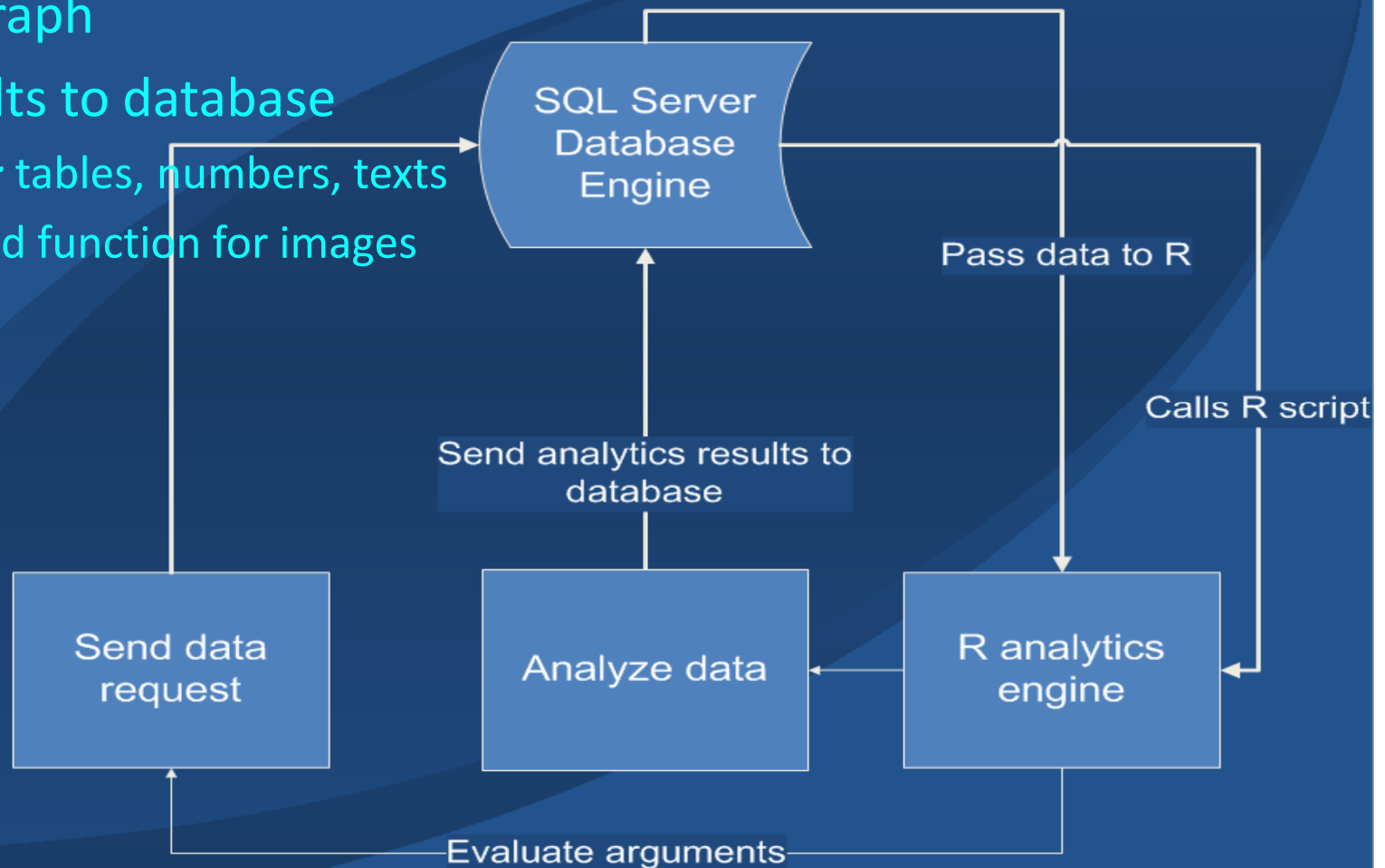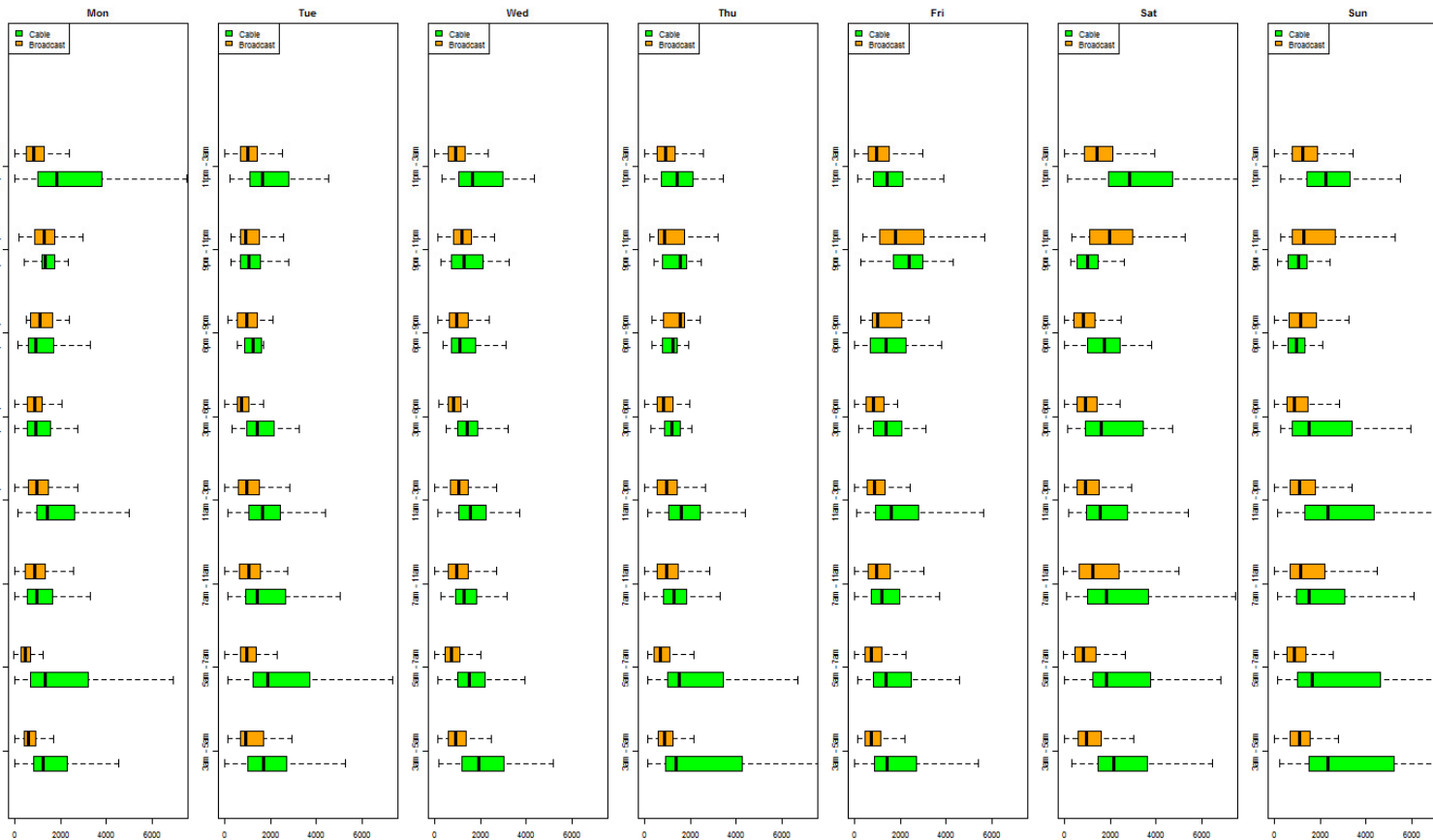

Proceed

Case Study – Data Visualization

Report url:
http://fathomdev66/Reports_DEV/Pages/Report.aspx?ItemPath=%2fuseR2010%2fboxplot

Case Study – Forecasting Model

www.newamerica.net

- Media buyer: how can we quantitatively predict our revenue?

- Forecast problem
  - given a specific airing (e.g. an ad for product A showing on ESPN at 9:00am PST)
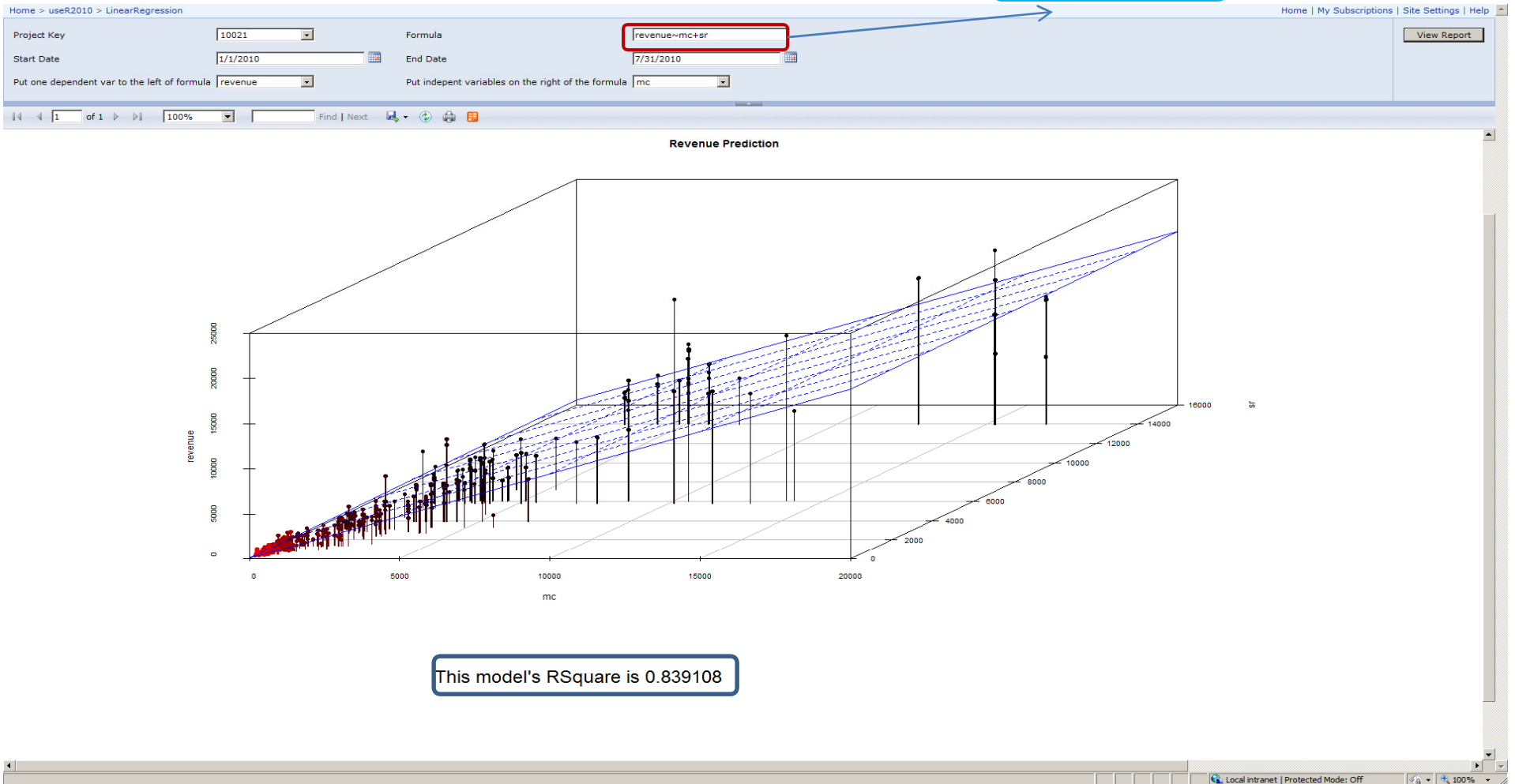
- Multiple regression

$$y_i = \beta_0 + \beta_1 x_{1i} + \cdots + \beta_p x_{pi} + \varepsilon_i$$

Case Study – Forecasting Model

- ## A real time regression tool

Regression Formula
Revenue ~ mc + sr

This model's RSquare is 0.839108

![Proceed logo]

Case Study – Publishing Forecasting Results

- ## R script snippet
  - ### Saving forecasted results

```
sqlSave(odbc.connect, forecast.result, tablename = 'Model10Forecast',
        append=TRUE, rownames=FALSE)
```

The R code inserts the forecasted results data back to the database using sqlSave function from the RODBC library:

forecast.result is the table that have our forecasted results,

tablename is what my target table in the database,

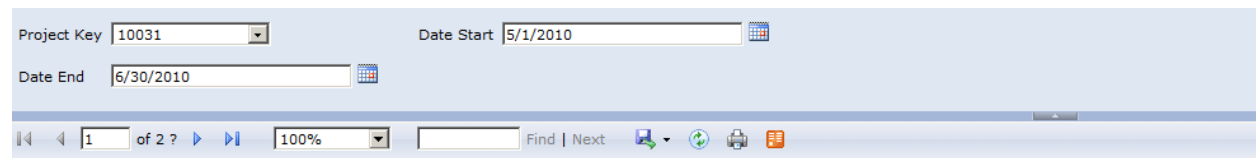*append* means we are going to add those rows to the database instead of overwriting the table,

rownames is the rownames of the result table, our target table doesn't have this column so we get rid of it.

See ?sqlSave for detailed explanation of this function.

Proceed

Case study – Publishing Forecasting Results

- ## After a good model is carefully selected, we can deploy it into production which updates daily

| Project Key | 10031 | | Date Start | 5/1/2010 | |
| Date End | 6/30/2010 | | | | |

Spot ID	startdatetime	CI_lower	predicted revenue	CI upper
595385	6/29/2010 11:30:00 PM	257.41	308.97	360.53
595100	6/29/2010 10:30:00 PM	1252.83	1312.35	1371.87
595232	6/29/2010 9:00:00 PM	344.09	398.55	453.02
596256	6/29/2010 9:00:00 PM	133.44	188.12	242.80
595796	6/29/2010 8:30:00 PM	270.80	325.32	379.84
593172	6/29/2010 8:30:00 PM	1123.16	1177.65	1232.13
595615	6/29/2010 7:30:00 PM	277.92	324.50	371.07
595233	6/29/2010 7:00:00 PM	564.09	610.42	656.75
593169	6/29/2010 6:00:00 PM	413.48	459.30	505.11
595384	6/29/2010 5:30:00 PM	599.75	645.31	690.88
593170	6/29/2010 4:30:00 PM	1104.39	1150.05	1195.71
595794	6/29/2010 4:30:00 PM	392.94	438.81	484.68
595446	6/29/2010 4:00:00 PM	1094.90	1154.12	1213.35
596421	6/29/2010 4:00:00 PM	1464.56	1510.30	1556.04
594614	6/29/2010 4:00:00 PM	7959.29	8043.79	8128.28

Proceed

Summary

- Analytics problems solved
  - R provides powerful advanced analytics for analyzing DRTV data
  - Actionable plan for media buy
- Engineering problems solved
  - SQL Server, SSRS provides convenient mechanism in developing analytics products developed by R
  - Publish analytics results to web portal, expose reports to internal team or external clients

Thank You!

- The authors would like to thank Brian Burdick, Dyng Au and Amanda Powter for their great help, advice and discussions which make this work possible

- The authors can be contacted at:
    - lwei at lucidcommerce dot com

- R, SQL sample scripts and related DRTV marketing topics can be found here:
    - http://blog.lucidcommerce.com/

Proceed