

VDSSt Frühjahrstagung

Dynamische Auswertungen mit R - am Beispiel von R- Shiny-Apps des Amtes für Statistik Magdeburg

Anne Seehase

Amt für Statistik, Wahlen und demografische Stadtentwicklung

Landeshauptstadt Magdeburg

Julius-Bremer-Straße 10

39104 Magdeburg

Tel. (0391) 540-2496

Fax (0391) 540-2807

Email anne.seehase@stat.magdeburg.de

Nutzung von R im Amt für Statistik der Stadt Magdeburg

Seit wann?

- seit 2014 im Einsatz zur Berechnung der Bevölkerungsprognose
 - ab 2015 kontinuierliche Umstellung von Stata auf R
- Auswahl der Software hängt von Präferenzen der Mitarbeiter ab.

Was?

- Erstellung von Grafiken
- Datenaufbereitung und Auswertung von Umfragen
- Regression-, Korrelations- und Zeitreihenanalysen
- Automatisiertes Routing und Entfernungsermittlung
- Mikrosimulation zur Schulbezirksoptimierung

Womit?

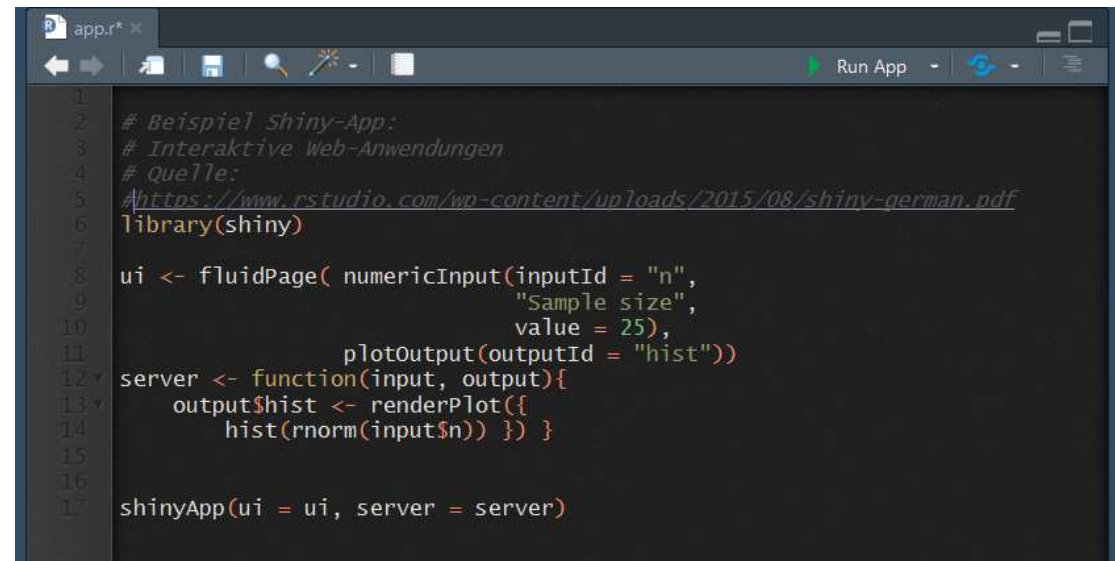
- R Studio (komplexe Auswertungen unter Verwendung von R-Projekten)
- Schnittstellen (API) zur OpenStreetMap, Google-Maps
- R-Markdown
- R-Shiny

R-Shiny Apps

- Ermöglicht die Erstellung von interaktiven Web-Anwendungen, R-Markdown-Dokumenten sowie Dashboards mittels R
- Voraussetzung zur Programmierung von R-Shiny-Apps ist das R-Pakete **shiny** (Basis)

Grundaufbau einer R-Shiny-App

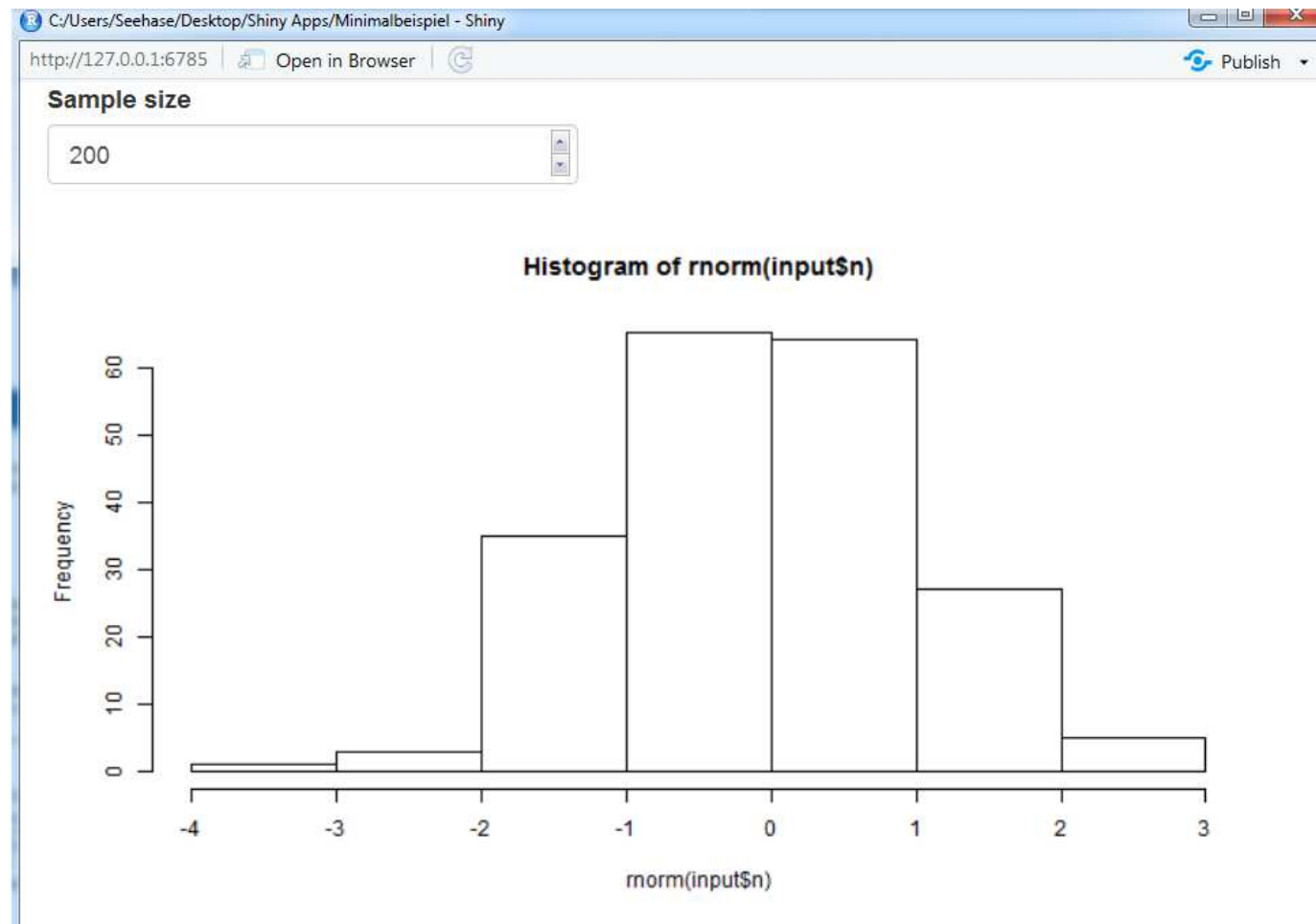
- **ui – User Interface**
 - Verschachtelte R-Funktionen zur Erzeugung einer Benutzeroberfläche
- **server - Programmcode**
 - Enthält R-Code zur Umsetzung der in der UI ausgewählten Einstellungen.
- **shinyApp**
 - Verbindung von ui und server-Elementen zu einer funktionierenden Anwendung



```
1 # Beispiel Shiny-App:
2 # Interaktive Web-Anwendungen
3 # Quelle:
4 # https://www.rstudio.com/wp-content/uploads/2015/08/shiny-german.pdf
5 library(shiny)
6
7
8 ui <- fluidPage( numericInput(inputId = "n",
9                           "Sample size",
10                          value = 25),
11                 plotOutput(outputId = "hist"))
12 server <- function(input, output){
13   output$hist <- renderPlot({
14     hist(rnorm(input$n)) }) }
15
16
17 shinyApp(ui = ui, server = server)
```

→ Ausführung ist sowohl lokal als auch online möglich.

UI - Minimalbeispiel



Zeitreihenapp

Hintergrund

- Wunsch, Komponentenzzerlegung für die Arbeitsmarktzahlen der Stadt Magdeburg durchzuführen
 - Darstellung der Trendkomponente
 - Analyse der Saisonkomponente
- Quartalsweise wiederkehrende Aufgaben
- Interesse auch in anderen Themenbereichen mit Zeitreihen
 - Unterjährige Entwicklung des Fremdenverkehr
 - Unterjährige Bevölkerungsentwicklung
- Divergente Kenntnisstand im Umgang mit Statistiksoftware

Ursprüngliche Quartalsweise Ermittlung der saisonbereinigte Zeitreihe

1. Berechnung in Excel

Jahr	Monat	Arbeitsmarktzahlen gesamt	gleitender Durchschnitt 11	Trendkomponente 2B	gleitende Saisonkomponente	Saisonkorrigierte Zeitreihe	Minus[1]	Auslage [1]	Saison	6-Schritt-Vorhersage	Trend [2]	Halt-Winter-Verfahren
2009	Januar	15.909				14.866	14.866	-230	1,07	15.909		2009
2009	Februar	16.175			1,07	15.096	15.004	-219	1,08	16.175		
2009	März	16.553			1,05	15.805	15.601	-194	1,06	16.553		
2009	April	16.592			1,02	16.236	16.070	-174	1,03	16.592		
2009	Mai	16.517			1,00	16.497	16.377	-160	1,01	16.517		
2009	Juni	16.439			0,98	16.942	16.797	-142	0,99	16.439		
2009	Juli	16.425	16.140,42	1,04	1,00	16.786	16.760	-139	1,00	16.425	16.140,42	
2009	August	16.552	16.188,33	1,02	1,01	16.426	16.465	-144	1,01	16.552	16.188,33	
2009	September	16.107	16.196,04	0,99	0,97	16.526	16.495	-139	0,98	16.107	16.196,04	
2009	Oktober	15.124	16.151,75	0,94	0,94	16.040	16.101	-146	0,94	15.124	16.151,75	
2009	November	15.941	16.059,67	0,94	0,93	16.199	16.150	-141	0,93	15.941	16.059,67	
2009	Dezember	15.270	15.926,92	0,96	0,96	15.970	15.978	-141	0,96	15.270	15.926,92	
2010	Januar	16.431	15.782,58	1,05	1,05	15.541	15.600	-149	1,07	16.431	15.782,58	2010
2010	Februar	16.403	15.642,42	1,06	1,05	15.495	15.411	-150	1,08	16.403	15.642,42	
2010	März	16.310	15.494,96	1,05	1,05	15.573	15.350	-147	1,06	16.310	15.494,96	
2010	April	15.772	15.359,17	1,03	1,03	15.433	15.261	-145	1,03	15.772	15.359,17	
2010	Mai	15.127	15.184,92	1,00	1,00	15.109	15.022	-148	1,01	15.127	15.184,92	
2010	Juni	14.843	14.944,17	0,99	0,99	15.114	14.962	-146	0,99	14.843	14.944,17	
2010	Juli	15.157	14.731,17	1,03	1,03	15.122	15.042	-139	1,01	15.157	14.731,17	
2010	August	14.556	14.551,71	1,02	1,02	14.743	14.803	-142	1,00	14.556	14.551,71	
2010	September	14.214	14.360,38	0,99	0,99	14.635	14.611	-143	0,98	14.214	14.360,38	
2010	Oktober	13.708	14.155,13	0,97	0,97	14.538	14.568	-140	0,94	13.708	14.155,13	
2010	November	12.295	13.983,71	0,88	0,88	13.224	13.433	-170	0,92	12.295	13.983,71	
2010	Dezember	12.258	13.858,33	0,88	0,88	12.820	12.913	-181	0,95	12.258	13.858,33	
2011	Januar	14.531	13.727,83	1,06	1,06	13.578	13.451	-159	1,08	14.531	13.727,83	2011
2011	Februar	14.296	13.610,08	1,06	1,06	13.406	13.349	-157	1,08	14.296	13.610,08	
2011	März	13.925	13.521,79	1,03	1,03	13.296	13.122	-159	1,06	13.925	13.521,79	
2011	April	13.231	13.447,25	0,98	0,98	12.947	12.834	-163	1,03	13.231	13.447,25	
2011	Mai	13.584	13.446,17	1,01	1,01	13.537	13.302	-144	1,02	13.584	13.446,17	
2011	Juni	13.407	13.540,33	0,99	0,99	13.652	13.443	-136	1,00	13.407	13.540,33	
2011	Juli	13.461	13.660,08	0,99	0,99	13.430	13.349	-135	1,01	13.461	13.660,08	
2011	August	13.724	13.778,46	1,00	1,00	13.622	13.584	-123	1,01	13.724	13.778,46	
2011	September	13.275	13.896,83	0,96	0,96	13.620	13.571	-120	0,98	13.275	13.896,83	
2011	Oktober	12.908	14.025,88	0,92	0,92	13.690	13.665	-114	0,94	12.908	14.025,88	
2011	November	13.069	14.122,08	0,93	0,93	14.056	14.133	-96	0,92	13.069	14.122,08	

2. Erstellung der Standardgrafik mit Stata

```

1 #delimit ;
2 /*generate achse = _n;
3 generate achse1 = achse - 0.2;
4 generate achse2 = achse + 0.2;*/
5 /* Quartalsberichtsgrafik 2-Jahres-Monatswerte*/
6 /* Achtung, Bezugspunkt der Monatswerte anpassen, Datengrundlage Grafik_Entwicklung_Stata.xl
7 twoway (bar gesamt_01 maennlich_01 achse1, barwidth(0.35 0.35) bcolor(maroon navy))
8       (bar gesamt_02 maennlich_02 achse2, barwidth(0.35 0.35) bcolor(maroon navy))
9       (scatter gesamt_s achse2, mcolor(black) msize(small)),
10
11
12 /* Veränderung der y-Achsen */
13 ylabel(0(2000)18000, grid gmax gmin angle(0)format(%9,0f) )
14 ytick(0(1000)18000, grid)
15
16 xlabel( 1 "Jan" 2 "Feb" 3 "März" 4 "April" 5 "Mai" 6 "Juni" 7 "Juli" 8 "Aug" 9 "Sep" 10 "Okt"
17
18 text(-2500 0.8 "2016", size(small) orientation(vertical))
19 text(-2500 1.2 "2017", size(small) orientation(vertical))
20
21 text(-2500 1.8 "2016", size(small) orientation(vertical))
22 text(-2500 2.2 "2017", size(small) orientation(vertical))
23
24 text(-2500 2.8 "2016", size(small) orientation(vertical))

```

Zeitreihenapp



Auswahl der Datei

Wähle .xlsx-Datei

Browse... Arb2018.xlsx

Upload complete

Verfügbarkeit der Daten (work in progress)

monatlich

Hilfe

Copy Print CSV

Jahr	Monat	Monat.Nr	
1990	Juli	7	3
1990	August	8	5
1990	September	9	6
1990	Oktober	10	8
1990	November	11	9
1990	Dezember	12	10
1991	Januar	1	11
1991	Februar	2	11
1991	März	3	11
1991	April	4	11
1991	Mai	5	12
1991	Juni	6	13
1991	Juli	7	16
1991	August	8	17
1991	September	9	17
...

Ziel: Vereinfachung der Handhabung zur (grundlegenden) Zeitreihenanalyse bei aggregierten Daten um wiederkehrende Bedarfe aber auch Sonderauswertungen abzudecken

- Ermöglicht Berechnungen mit Zeitreihen für beliebige Datensätze
 - Komponentenzerlegung
 - Trendschätzung (Holt-Winters-Verfahren)
 - Autokorrelationsanalyse
- Darstellung von Interaktiven Plot und deren Download
- Erstellung von Standardplots mit PDF-Downloads (in Arbeit)

Weiterentwicklung: weitere Flexibilisierung des Tools, um Grafiken für Standardlayouts auch für andere Themenbereich außer Arbeitsmarkt darstellen zu können

Tourenplanung zur Europa- und Kommunalwahl

Hintergrund

- In Rahmen der Wahlen müssen 159 Wahllokale in 93 Einrichtungen anlässlich der Ausstattung mehrfach angefahren werden.
 - An- und Abtransport von Wahlspezifischen Gegenständen
 - Abnahme der Einrichtung der Wahllokale durch Mitarbeiter der Briefwahlstelle
 - Anlieferung des Wahlkoffers und der Stimmzettel am Wahlsonntag
- Realistische Abschätzung von Fahrtzeiten zwischen den Objekten im Rahmen der Tourenplanung schwierig
- Zusätzlich Sonderwünsche der Einrichtung bzgl. Abnahmezeit/Transportzeitraum aufgrund äußerer Gegebenheiten

Ziel: Tool zur Unterstützung der Tourenplanung unter Berücksichtigung der Möglichkeiten von R

- Google-Maps-Direction-API zur automatisierten Abfrage von Reisezeit (Durchschnitt und unter Verkehrsbedingungen zwischen Adresse)
- Verfahren der mathematischen Optimierung zur Tourenplanung → Travel-Salesman-Problem zur Ermittlung eines optimalen Rundweg zwischen verschiedenen Objekten (heuristische Lösung mittels TSP-Paket in R)
- Möglichkeit, diesen Rundweg auf Basis verschiedener Maximal-Kriterien zu partitionieren und somit Teilabschnitte zu ermitteln.

Routenabfrage Google-Maps-Direction-API

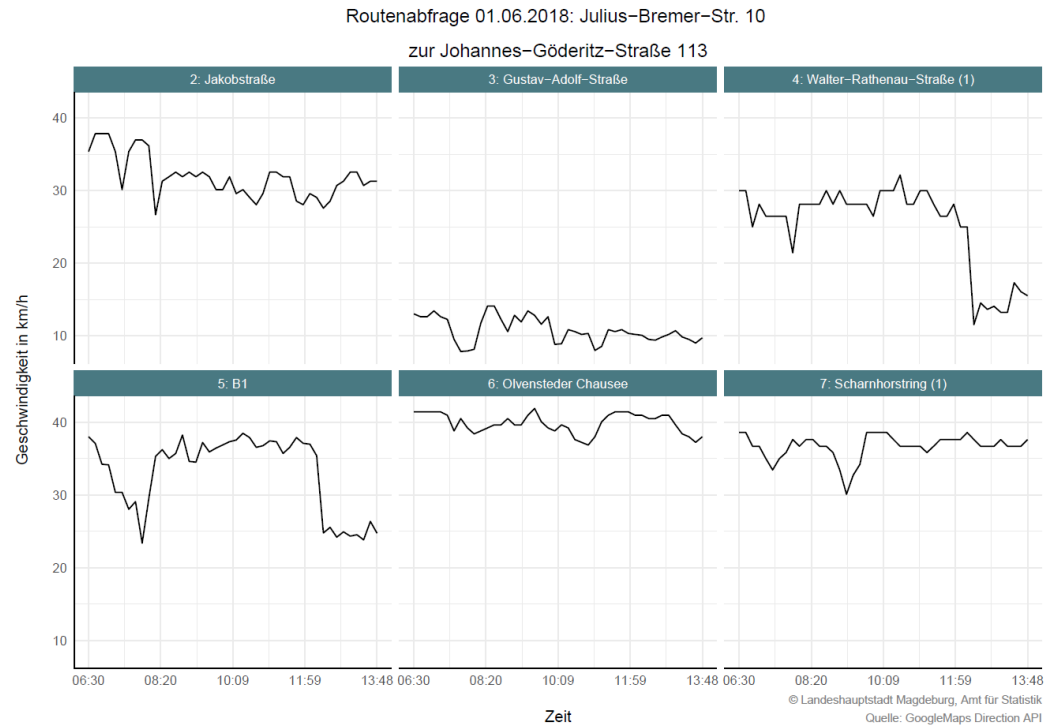
- Vorbereitende Abfrage der Routen von jeder Einrichtung zu jeder Einrichtung plus der Adresse des Startpunktes (Lager) der Routen
- Abfrage liefert neben der Route auch die Entfernung sowie durchschnittliche Reisezeit und Reisezeit unter aktuellen Verkehrsbedingungen

```

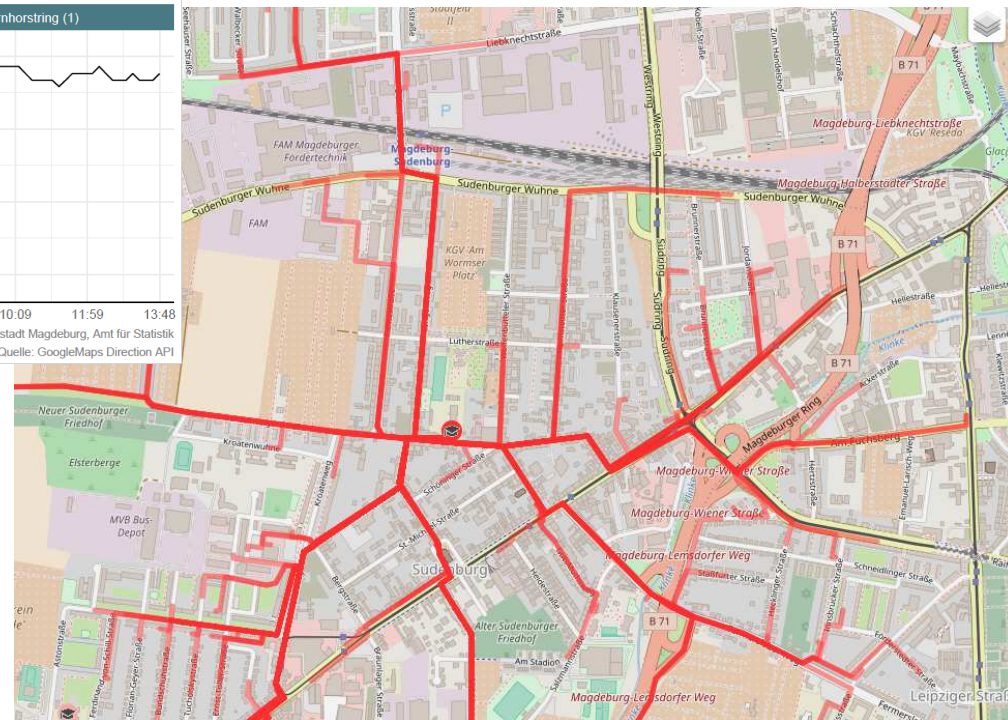
10 library(httr)
11 library(rvest) # Anfragen
12 library(mapsapi)
13 library(sf) # Umgang mit sfc_Coordinates-String
14
15 Map.Api.Request<-function(Start=NULL,
16                           Ziel=NULL,
17                           key=NULL # Schlüssel für Map.Api ergänzen!
18 ) {
19
20   Pfad1<-"https://maps.googleapis.com/maps/api/directions/xml?origin="
21   Key=key #
22   Anfrage<-paste(Pfad1,Start,"&destination=",Ziel,"&key=",Key,"&mode=driving&alternatives=true&departure_time=now",sep="")
23   out<-read_xml(Anfrage)
24   return(out)
25 }

```


Alternative Routenabfragen



- Anfragen an Google-Maps-Direction-API erfordern Anmeldung (API-Key)
- Anfragen mittlerweile kostenpflichtig (20.000 Anfragen im Monat frei)
- Vorteil: Reisezeiten zu Verkehrslagen



- Alternativ Routinganfragen auf Open-Streetmap-Server
- Lokaler Spiegel der OSM möglich
- R-Package OSRM
- Vorteil: Sperrung bestimmter Strecken

Lösung des Sales-Man-Problem mit TSP-Package

- Ermittelt den „optimalen“ (kürzesten) Rundweg zwischen einer Reihe von Punkten
- Basis bildet symmetrische Distanzmatrix
- Verschiedene Lösung-Methoden (heuristisch, analytisch)

```
library(openxlsx)
library(sp)
library(TSP)

Input <- matrix(0, nrow(Abfrage0$Distanz[-Zentrum,]), nrow(Abfrage0$Distanz[-Zentrum,]))
Input[lower.tri(Input)] <- t(Abfrage0$Distanz[-Zentrum, -Zentrum])[lower.tri(t(Abfrage0$Distanz[-Zentrum, -Zentrum]))]
Input[upper.tri(Input)] <- Abfrage0$Distanz[-Zentrum, -Zentrum][upper.tri(Abfrage0$Distanz[-Zentrum, -Zentrum])]
# Package TSP
# TSP - Algorithmus : in der hier verwendeten Methode handelt es sich um ein heuristisches V
tsp <- TSP(Input)
tour <- solve_TSP(tsp)
```



Partitionierung des Rundweges

- Basis bilden „Ladepkapazitäten“
- Wobei Ladepkapazitäten auch einfach eine vordefinierte Anzahl von Wahllokalen sein kann
- Integration des Vorgehen in einer Shiny-App ermöglicht Anwendung von Mitarbeitern, welche keine Kenntnis von R-Programmierung haben

App zur Tourenplanung

- Berücksichtigung unterschiedlichen Tourenarten
- Export von Tourenplänen mit Reisezeiten
- Derzeit in Erprobungsphase

Datei Bearbeiten Ansicht Favoriten Extras ?

Tourenplanung Europa und Kommunalwahl 26.05.2019

Tourart:
Dauereinlagerung

Maximale Beladung Urnen:
24

Maximale Beladung Wahlkabine:
40

Zeitpuffer Halt:
10

Zeitraum 1 Zeitraum 2 Zeitraum 3 Zeitraum 4 Zeitraum 5 Kontrolle

Touren

Copy Excel CSV PDF Print

Rundweg

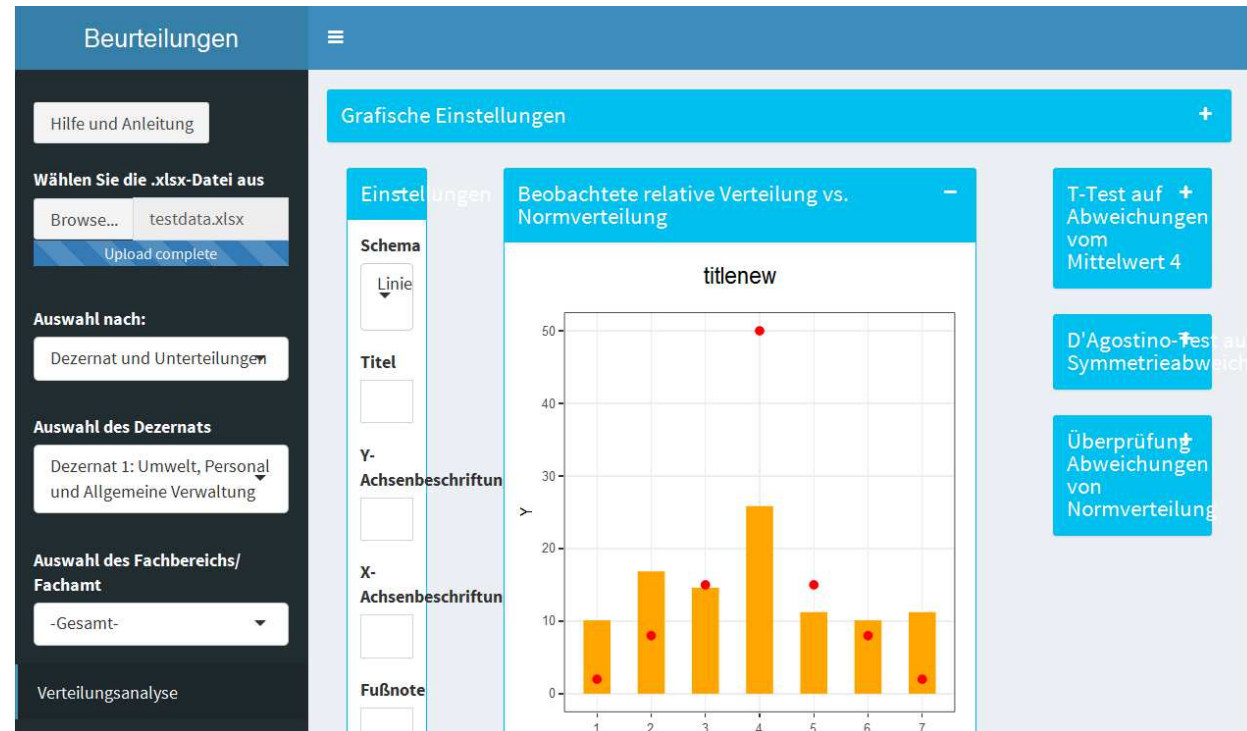
Aktualisiere Rundweg

Search:

Tour	Objekt.ID	Zeit_h	Zeit_Verkehr_h	Bezeichnung	Adresse	HNR	HNRA	Anzahl	Barrierefreiheit	Wahlurnen	Standwahlkabinen	Lampe	Klingel	Vorhang	
37	Tour 001	001	00:16	00:16	Sportgymnasium Magdeburg	Friedrich-Ebert-Straße	16	NA	2	barrierefrei	4	2	0	1	0
38	Tour 001	002	00:27	00:27	Sportsek. "Hans Schellheimer"	Friedrich-Ebert-Straße	51	NA	2	barrierefrei	4	2	0	0	0
39	Tour 001	003	00:42	00:43	PSV MD 1990 e.V.-Stadion Neue Welt	Berliner Chaussee	219	NA	1	barrierefrei	1	0	0	0	0
41	Tour 001	004	00:57	00:57	BürgerBüro Ost	Tessenowstraße	15	NA	1	barrierefrei	1	0	0	0	0

Verteilungsvergleiche für Personalbereich

- Mitarbeiterbewertung sollte einer bestimmten Verteilung folgen, welche sich in der Mitte (Note 4, entspricht 100 %) orientiert
- Aufgabe der Personalabteilung ist es zu prüfen, ob bestimmten Organisationseinheiten nicht zu gut oder zu schlecht bewertet
- Ausreißer sollten begründet werden



- Entwicklung einer Anwendung, welche nach Einlesen des Datensatzes eine automatisierte Prüfung der abgegebenen Bewertungen nach bestimmten Auswertungsgruppen ermöglicht.

Fazit und Ausblick

- Shiny-Apps bilden gutes Werkzeug, um dynamische Auswertungen einem breiten Personenkreis zu ermöglichen.
- Für Routineaufgabe wird somit ein R-basierte Lösung auf Knopfdruck möglich.
- Integration von bestehenden R-Code ist einfach, sofern man das ganze unter der Logik von Funktionen betrachtet: Input → Ausführung → Output
- Möglichkeit, R-Shiny-Apps unabhängig über die Installation von R zu verwirklichen durch Hosten auf einen Server
 - Server erfordern das Betriebssystem Ubuntu
 - Hosten auf den Shiny-Server ist ab gewissen Umfang kostenpflichtet, Datensicherheit?
- Möglichkeit zur lokalen Anwendung von R-Shiny-App (Bedingung: Installation von R-Studie)
- Integration einer lokalen Version in eine .exe Datei (In Prüfung)
- Weitere Integration von R-Shiny-Apps im Amt für Statistik bzw. als Dienstleistung für andere Fachbereich geplant

otto präsentiert otto stellt vor otto informiert otto gibt bekannt otto zeigt

Vielen Dank für Ihre Aufmerksamkeit!

Anne Seehase
Amt für Statistik, Wahlen und demografische Stadtentwicklung
Landeshauptstadt Magdeburg
Julius-Bremer-Straße 10
39104 Magdeburg
Tel. (0391) 540-2496
Fax (0391) 540-2807
Email anne.seehase@stat.magdeburg.de