

Inhaltsverzeichnis

1	Propensity Score Methods I	1
1.1	Vorbereitung	1
1.2	Grundidee	2
1.3	Schätzung	3
1.3.1	Mehrschrittig mit <code>glm()</code>	3
1.3.2	Integriert in <code>MatchIt</code>	4
1.4	Überprüfung	5
1.4.1	<code>summary()</code> Optionen	5
1.4.2	<code>plot()</code> Optionen	5
1.4.3	Overlap im PS	6
1.4.4	Balance auf den Kovariaten	8
1.4.5	Anpassung des Modells	10
1.4.6	QQ-Plots der Kovariaten	11

1 Propensity Score Methods I

Lernziele und Vorgehen

1. Schätzung der Propensity Scores
 - a) beispielhaft über logistisches Regressionsmodell (`glm()`)
 - b) integriert in `matchit()`
2. Prüfung des Overlap (Verteilung des PS) zwischen Gruppen
3. Prüfung der Balance (Verteilung der Kovariaten) zwischen Gruppen
4. ggfs. Anpassung des Modells

Es ist wichtig zwei Prozesse zu unterscheiden: Die Schätzung des Propensity Scores und der Matching-Prozess kann mehrfach angepasst werden (Sitzung 3), **nicht jedoch** die Schätzung des Treatment-Effekts (Sitzung 4, siehe zum Beispiel [Data Dredging](#)).

1.1 Vorbereitung

- Wir setzen das Arbeitsverzeichnis und öffnen den Datensatz `bdi_data.rda`.
- Wir aktivieren das Paket `MatchIt`.
- Außerdem setzen wir zur Reproduzierbarkeit der Ergebnisse, die einen Zufallsprozess beinhalten, den Startwert des Zufallszahlengenerators über `set.seed()` fest.

```
setwd("C:/Users/me/myworkingdirectory")
load("bdi_data.rda")
library(MatchIt)
set.seed(12345) # Zur Vergleichbarkeit der Ergebnisse
```

- Wir arbeiten mit demselben Datensatz wie in der vorherigen Sitzung zum Kovariaten-Matching.
- In dem Skript zur 2. Sitzung finden sich auch die Deskriptivstatistiken nach Gruppen aufgeteilt.

Zur Erinnerung:

- Die Gruppenvariable X heißt hier **group** (**group==1** bedeutet, zusätzlich wurde bei den PatientInnen eine achtsamkeitsbasierte Intervention durchgeführt).
- Das Outcome Y ist der Depressionswert zum 2. Messzeitpunkt, **bdi2**.
- Mögliche Kovariaten Z_i bzw. **Z** sind
 - der Depressionswert vor der Intervention, (**bdi1**)
 - das Alter (**age**)
 - die Anzahl bisheriger Therapiesitzungen (**sess**)
 - eine Skala der kognitiven Leistungsfähigkeit (**cogn**), und
 - das Vorliegen einer Substanzabhängigkeit (**addic**; 0 = nein, 1 = ja)

1.2 Grundidee

Statt beim Kovariaten-Matching auf vielen Kovariaten zu matchen, wird pro Person als Linearkombination aller Kovariaten ein Wert, der *Propensity Score* (S) geschätzt und die Personen nach diesem gematcht. Dieser Matching-Prozess kann überprüft und wiederholt werden, bis ein zufriedenstellendes Ergebnis erreicht ist. Der Propensity Score (PS) ist für jede Person die geschätzte Wahrscheinlichkeit, zur Behandlungs- oder Kontrollgruppe zu gehören, basierend auf ihren Werten für die Kovariaten:

$$\pi = P(X = 1 | \mathbf{Z})$$

- π : Propensity Score
- $P(X = 1 | \mathbf{Z})$: Bedingte Wahrscheinlichkeit, in der Treatment-Gruppe zu sein, gegeben Kovariaten **Z**

Ziel: Es soll die Bedingung der **Strong Ignorability** erfüllt sein, das heißt, die *potential outcome* Ergebnisse Y^0 und Y^1 sollen unabhängig von der Treatmentzuweisung X sein, wenn π kontrolliert wird.

$$(Y^0, Y^1) \perp X | \pi$$

1.3 Schätzung

- Die Propensity Scores können mithilfe der logistischen Regression geschätzt werden.
- Zunächst verwenden wir die bekannte `glm()` Funktion, um die Schätzung manuell durchzuführen, und erhalten dann in einem zweiten Schritt die geschätzten Werte (= Propensity Scores).
- Als praktische Alternative zeigen wir das Vorgehen mit der Verwendung der `matchit()` Funktion.

1.3.1 Mehrschrittig mit `glm()`

- In der logistischen Regression sieht das Modell auf der rechten Seite der Regressionsgleichung wie in der multiplen Regression aus (nur ohne Fehlerterm).
- Außerdem wurde das X der Prädiktoren mit Z für Kovariaten ersetzt. Prädiktor ist die Bezeichnung der unabhängigen Variablen in der Regressionsgleichung, Kovariate ist die Bezeichnung der Rolle der Variablen im Modell, also Kontrollvariable, korrigierende Variable.
- Die linke Seite der Gleichung ist logit-transformiert, das bedeutet, sie repräsentiert die logarithmierten Odds (Verhältnis der Chancen) in der Treatmentbedingung zu sein vs. in der Kontrollbedingung zu sein, gegeben den Kovariaten.
- Das fett gedruckte \mathbf{Z} entspricht dabei allen Z -Variablen auf der rechten Seite der Gleichung (in Matrixschreibweise).

$$\ln \left(\frac{P(X = 1|\mathbf{Z})}{1 - P(X = 1|\mathbf{Z})} \right) = \beta_0 + \beta_1 Z_1 + \beta_2 Z_2 + \dots + \beta_k Z_k$$

- Um den Logit wieder in den Propensity Score – die Wahrscheinlichkeit $\pi = P(X = 1|\mathbf{Z})$ – umzurechnen, muss die rechte Seite der Gleichung umgeformt werden.

$$\pi = P(X = 1|\mathbf{Z}) = \frac{e^{\beta_0 + \beta_1 Z_1 + \beta_2 Z_2 + \dots + \beta_k Z_k}}{1 + e^{\beta_0 + \beta_1 Z_1 + \beta_2 Z_2 + \dots + \beta_k Z_k}}$$

- In R kann die logistische Regression berechnet werden mithilfe des `glm()`-Befehls (*generalized linear model*).
- Wir spezifizieren im Folgenden eine logistische Regression zur Vorhersage unserer Treatmentbedingung `group` durch die im Datensatz vorhandenen Kovariaten `bdi1`, `age`, `cogn`, `sess` und `addic`.

```
mod_ps <- glm(group ~ bdi1 + age + cogn + sess + addic,
              data = bdi_data,
              family = binomial)
```

- Das Argument `family` spezifiziert dabei die Verteilungsannahme für unsere abhängige Variable (hier: die Treatmentbedingung X). Die Option (`link = 'logit'`) ist hier weggelassen, da der Logit die Standardeinstellung für `binomial` ist.



- Die geschätzten Werte (Propensity Scores) und die dazugehörigen logits können pro Person extrahiert und zum Beispiel dem Datensatz hinzugefügt werden mit den Befehlen:
- `predict()`: gibt vorhergesagten Logit aus
- `fitted()`: gibt $\hat{P}(X = 1|\mathbf{Z})$ – die PS – aus

```
bdi_data$logit <- predict(mod_ps)
bdi_data$Pscore <- fitted(mod_ps)
head(bdi_data)
```

	bdi1	bdi2	age	sess	cogn	addic	group	logit	Pscore
1	43	27	28	24	94	0	1	0.30399116	0.5754179
2	33	33	41	16	73	0	1	0.05846983	0.5146133
3	35	25	25	13	85	0	1	-0.06506508	0.4837395
4	27	31	46	24	74	0	1	-0.25412944	0.4368074
5	32	33	28	19	81	0	1	-0.06243609	0.4843960
6	21	19	35	22	90	0	1	-1.01159627	0.2666676

- Für die ersten sechs Personen im Datensatz kann man sehen, dass alle in der Interventionsgruppe waren (`group == 1`), ihr PS aber variiert von 0.27 bis 0.58.
- Wahrscheinlichkeiten, die geringer als 0.5 sind, gehen mit einem negativen Logit einher.
- Anhand dieser `Pscore` Variable könnte man nun `matchen`. Wir nutzen im folgenden aber das Paket `MatchIt`, das Schätzung und Matching in einem Schritt erledigt.

1.3.2 Integriert in MatchIt



- Die Voreinstellung für das Distanzmaß in der Funktion `matchit()` lautet `distance = "logit"`, was dem gerade berechneten PS entspricht.
- Die vorherige Berechnung über `glm()` ist dabei nicht notwendig.
- Wir möchten nun Treatment- und Kontrollprobanden in unserem `bdi_data`-Datensatz anhand von Nearest Neighbor Matching auf Basis des Propensity Scores `matchen`. Dafür verwenden wir alle vorhandenen Kovariaten:

```
ps_mi <- matchit(group ~ bdi1 + age + cogn + sess + addic,
                 data = bdi_data,
                 method = "nearest",
                 distance = "logit")
```

1.4 Überprüfung

Es stehen für `matchit` Objekte zur Überprüfung von Overlap und Balance die `summary()` und `plot()` Funktionen mit folgenden Optionen zur Verfügung:

1.4.1 `summary()` Optionen

Der `summary()` Befehl gibt Kennwerte der Balance für gematchte und ungematchte Stichproben aus.

Argument	Bedeutung
<code>standardize = TRUE</code>	Standardisierte Mittelwertsdifferenzen etc.
<code>interactions = TRUE</code>	Kennwerte auch für Interaktionsterme berichten, die nicht unbedingt im Modell enthalten sind
<code>addlvariables = TRUE</code>	Kennwerte auch für Variablen berichten, die nicht im (PS-)Modell enthalten sind

1.4.2 `plot()` Optionen

Der `plot()` Befehl generiert Grafiken zur Beurteilung von Overlap und Balance.

Argument	Bedeutung
<code>type = "jitter"</code>	PS-Verteilung nach Treatment-Gruppen und Unmatched vs. Matched
<code>type = "hist"</code>	PS-Verteilung getrennt nach Gruppen und Raw vs. Matched
<code>type = "QQ"</code>	Gruppenspezifische Kovariaten-Verteilungen als QQ-Plot
<code>interactive = TRUE</code>	Ob die Grafik interaktiv angezeigt werden sollen
<code>discrete = 1</code>	Verhindern des <i>Jitterns</i> für dichotome Variablen
<code>which.xs =</code>	Auswählen welche Kovariaten angezeigt werden sollen
<code>subclass =</code>	Falls einzelne Subgruppen ausgewählt werden sollen (Stratifizierung, s.u.)

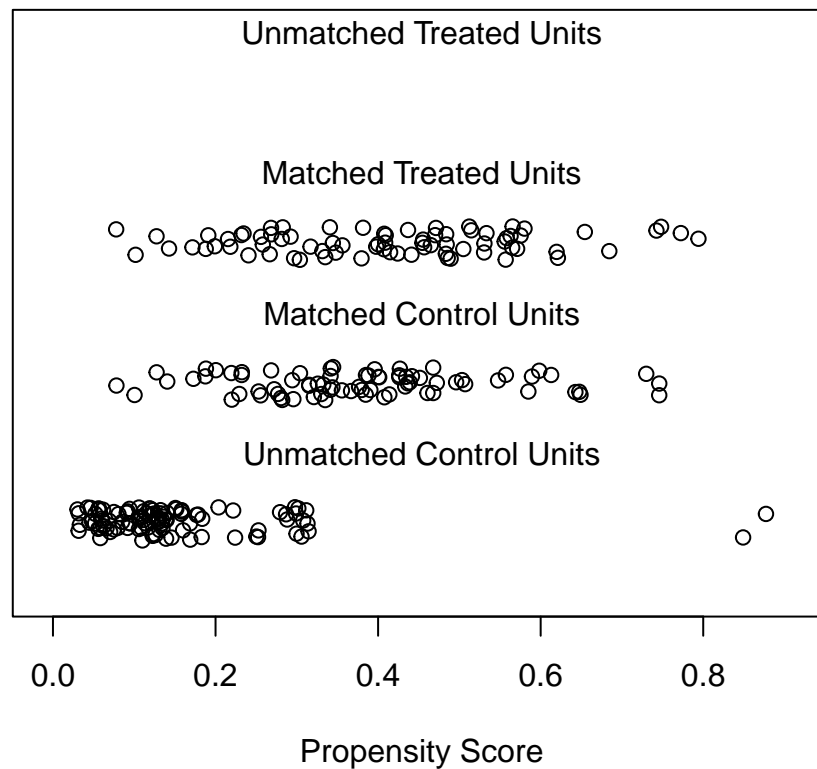
- `plot(summary(ps_mi, standardize = TRUE))`
 - Veränderung in der standardisierten Mittelwertsdifferenz durch Matching
- Ggfs. `layout(1)` ausführen, um die Darstellung eines einzelnen Plots wiederherzustellen (statt 4 in 1).

1.4.3 Overlap im PS

- Zunächst betrachten wir die Verteilung der Propensity Scores.

```
plot(ps_mi, type = "jitter", interactive = FALSE)
```

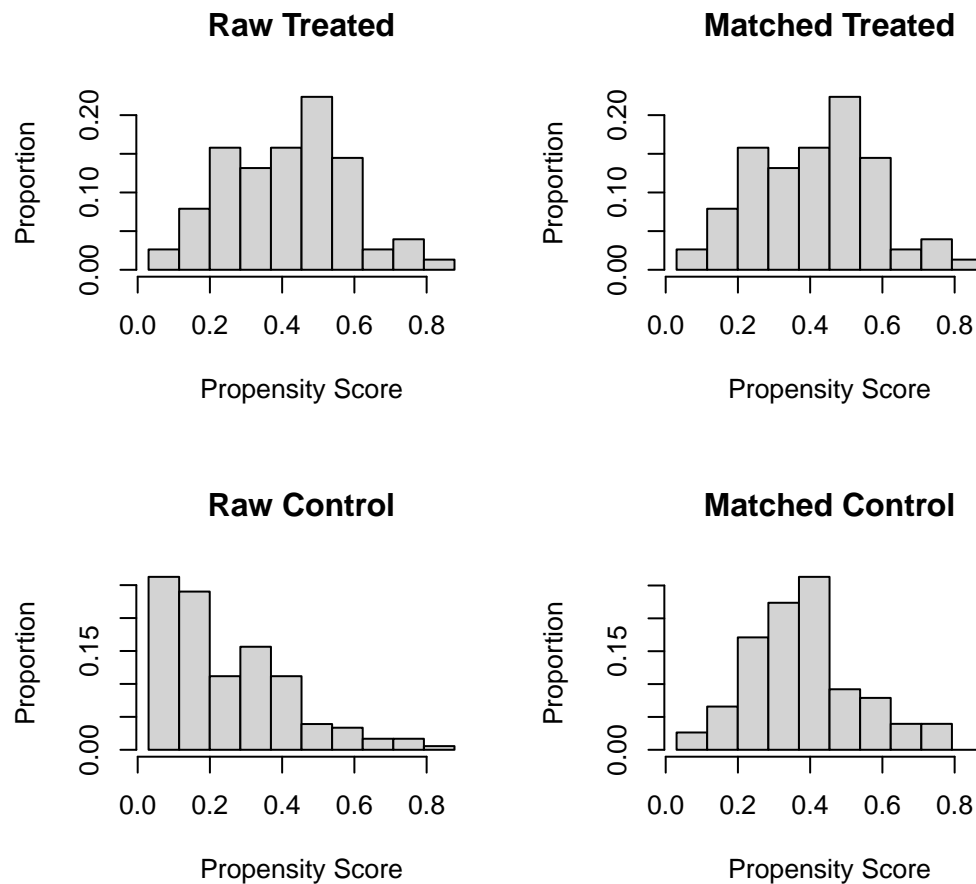
Distribution of Propensity Scores



- In dieser Abbildung sind die PS auf der X-Achse für Personen nach Gruppenzugehörigkeit dargestellt:
 - Personen in der Treatment-Gruppe, die nicht gematcht wurden (*Unmatched Treatment Units*, hier nicht vorhanden)
 - Personen in der Treatment-Gruppe, die gematcht wurden (*Matched Treatment Units*, hier alle)
 - Personen in der Kontroll-Gruppe, die gematcht wurden (*Matched Control Units*)
 - Personen in der Kontroll-Gruppe, die nicht gematcht wurden (*Unmatched Control Units*)

- Von besonderem Interesse ist also der Vergleich der mittleren beiden Gruppen (die gematchte Stichprobe).
- Zunächst einmal ist festzustellen, dass es über alle Gruppen hinweg starke Unterschiede im PS zwischen Personen gibt (Streubereich 0.03 bis 0.88).
- Personen mit sehr hohen (>0.8) und sehr niedrigen (ca. <0.08 , s.u.) Propensity Scores kommen lediglich in der Kontrollgruppe vor. Diese Bereiche des PS fallen somit nicht in die *region of common support* und die entsprechenden Personen wurden nicht gematcht.
- Als nächstes visualisieren wir uns die Verteilung der Propensity Scores in der Gesamtstichprobe (Raw) sowie der Stichprobe der gematchten Personen (Matched) getrennt nach Treatment-Gruppen anhand eines Histogramms:

```
plot(ps_mi, type = 'hist')
```



- Man erkennt, dass sich die Verteilung der Treated (1. Zeile) nicht verändert hat, da niemand ausgeschlossen wurde.

- Die Verteilung des Propensity Scores in der Kontrollgruppe ist nach dem Matching weniger linkssteil und hat sich der Verteilung der Treatmentbedingung angenähert.
- Falls man gerne die Verteilungskennwerte der PS pro Gruppe nach dem Matching erhalten möchte, bietet es sich an, die Zusammenfassung für den PS getrennt nach der Gruppe anzufordern. Der Variablenname, der von `matchit()` für die PS verwendet wird, ist `distance`.

```
aggregate(distance ~ group, data = match.data(ps_mi), summary)
```

```
group distance.Min. distance.1st Qu. distance.Median distance.Mean
1      0    0.07807280      0.28170250      0.37820580      0.38272330
2      1    0.07774842      0.28237545      0.41208270      0.41378672
distance.3rd Qu. distance.Max.
1          0.45372454      0.74588975
2          0.53050748      0.79412865
```

- Im unteren Bereich der Verteilung (Minimum und 1. Quartil) sind sich die Verteilungen sehr ähnlich, ab dem Median hat die Treatment-Gruppe aber jeweils (leicht) höhere Werte.

1.4.4 Balance auf den Kovariaten

Für eine Zusammenfassung der Balance auf den Kovariaten lassen wir uns die `summary()` des `matchings` ausgeben.

- Richtwerte für standardisierte Mittelwertsunterschiede:
 - $< .25$ SD (z. B. Stuart & Rubin, 2007)
 - $< .10$ SD (z. B. Steiner et al., 2010)



```
summary(ps_mi, standardize = TRUE, improvement = TRUE)
```

Summary of balance for all data:

	Means Treated	Means Control	Std. Mean Diff.
distance	0.4137867	0.248895	1.008729786
bdi1	30.9605263	20.955307	1.037501693
age	35.8157895	33.832402	0.284879230
cogn	82.7368421	91.960894	-0.890296256
sess	20.2763158	15.363128	0.476098424
addic	0.4210526	0.424581	-0.007146393

Im ersten Abschnitt (`Summary of balance for all data`) sehen wir die Kennwerte für die Gesamtstichprobe vor dem Matching. Für die PS haben wir eine standardisierte Mittelwertsdifferenz (`Std. Mean Diff.`) von ≈ 1 . Die Spalten, die mit `eCDF` beginnen, sind nicht abgebildet und werden im Rahmen der QQ-Plots erörtert.

`Summary of balance for matched data:`

	Means Treated	Means Control	Std. Mean Diff.
<code>distance</code>	0.4137867	0.3827233	0.190031352
<code>bdi1</code>	30.9605263	29.3947368	0.162366183
<code>age</code>	35.8157895	35.8815789	-0.009449519
<code>cogn</code>	82.7368421	83.4605263	-0.069849277
<code>sess</code>	20.2763158	20.0526316	0.021675481
<code>addic</code>	0.4210526	0.4473684	-0.053300179

Im zweiten Abschnitt (`Summary of balance for matched data`) sehen wir die Kennwerte für die gematchte Stichprobe. Die standardisierte Mittelwertsdifferenz (`Std. Mean Diff.`) beträgt bis zu 0.19 auf dem PS und 0.16 auf den Vortestwerten `bdi1`. Nach dem strengen Kriterium von Steiner wäre dies zu hoch.

`Percent Balance Improvement:`

	Std. Mean Diff.	Var. Ratio
<code>distance</code>	81.16132	-13.145905
<code>bdi1</code>	84.35027	71.039105
<code>age</code>	96.68297	-17.173779
<code>cogn</code>	92.15438	58.476748
<code>sess</code>	95.44727	1.256109
<code>addic</code>	-645.83333	NA

Im unteren Abschnitt werden die beiden vorherigen verglichen und die Verbesserung in Prozent angegeben. So hat sich die absolute Ungleichheit auf `age` von 0.285 (all data) auf 0.009 (matched data) um rund $(0.285-0.009)/0.285 = 96.68\%$ verbessert.

Die extrem hohe Verschlechterung bei `addic` ist darauf zurückzuführen, dass der Unterschied vor dem Matching quasi 0 war (und an diesem ja relativiert wird). Außerdem handelt es sich um ein dichotomes Merkmal, so dass die Mittelwerte als relative Häufigkeiten interpretiert werden können.

`Sample Sizes:`

	Control	Treated
All (ESS)	179	76
All	179	76
Matched (ESS)	76	76
Matched	76	76
Unmatched	103	0
Discarded	0	0

Der letzte Abschnitt gibt eine Übersicht der Häufigkeiten pro Gruppe von ursprünglich enthaltenen (All), (un-)gematchten (Matched und Unmatched) und ausgeschlossenen Personen (Discarded; falls bei `matchit()` das Argument `discard` spezifiziert wurde).

1.4.5 Anpassung des Modells



Falls wir das strenge Kriterium anwenden, können wir mit der Balance noch nicht zufrieden sein. Um die Schätzung des PS zu ändern, kann das Modell, das zur Schätzung genutzt wird, um Variablen oder Terme erweitert werden. Da wir bereits alle Variablen im Datensatz im Modell haben, bleibt uns noch die Möglichkeit, nach Termen zu schauen, die wir in den PS einfließen lassen wollen. - Man könnte über den `caliper` auch strengere Matching-Kriterien definieren, riskiert dabei aber einen Verlust an Personen. - In unserem Fall ist es ausreichend, denn Matching-Algorithmus zu ändern auf `optimal` (Paket `optmatch` muss installiert sein):

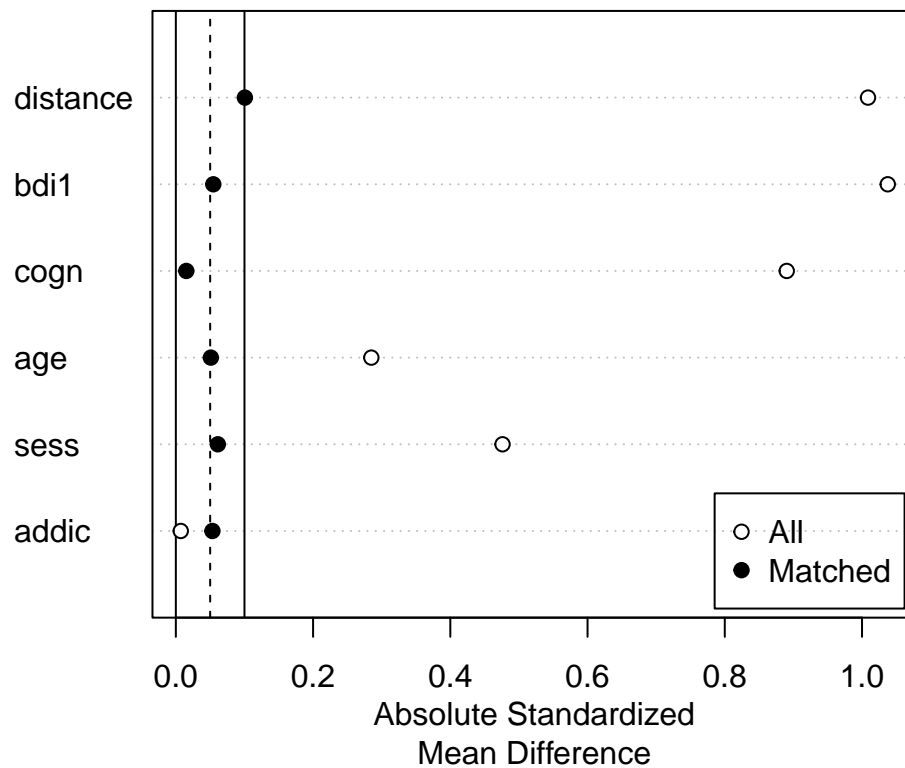
```
ps_mi2 <- matchit(group ~ bdi1 + cogn + age + sess + addic, data = bdi_data,
                  method = "optimal",
                  distance = "logit")
summary(ps_mi2, standardized = TRUE)
```

Summary of balance for matched data:

	Means Treated	Means Control	Std. Mean Diff.	Var. Ratio
distance	0.4137867	0.3973432	0.10059374	0.9634810
bdi1	30.9605263	30.4342105	0.05457687	1.0304654
cogn	82.7368421	82.8947368	-0.01523984	0.6461702
age	35.8157895	36.1710526	-0.05102740	0.8926150
sess	20.2763158	19.6447368	0.06120136	0.6976079
addic	0.4210526	0.4473684	-0.05330018	NA

Die Veränderung der standardisierten Mittelwertsunterschiede lässt sich gut darstellen über die `plot()` Funktion, angewendet auf die `summary()` des `matchit` Objekts:

```
plot(summary(ps_mi2, standardize = TRUE))
```



1.4.6 QQ-Plots der Kovariaten

- In der `summary()` haben wir uns bisher v.a. auf die *mittleren* Unterschiede konzentriert.
- Die Spalten mit `eCDF` bzw. im unstandardisierten Fall `eQQ` beziehen sich auf andere Punkte der Verteilung, die sich gut in sog. QQ-Plots abbilden lassen.
- Hierbei sind die Quantile der Kontrollgruppe auf der *X*-Achse gegen die Quantile der Treatment-Gruppe auf der *Y*-Achse abgetragen. Bei gleicher Verteilung befinden sich alle Punkte auf der diagonalen Linie.
- Wir betrachten diesen Plot für die Kovariaten `bdi1`, `age` und `cogn`. Im Gegensatz zu oben ist dies ein Ausschnitt der unstandardisierten Ausgabe.
- Beispielsweise beträgt der unstandardisierte Unterschied der Maxima von `bdi1` zwischen den Gruppen 4 Punkte, der entsprechende Punkt liegt abseits der Diagonalen, aber noch innerhalb des angezeigten Toleranzbereiches.

```
summary(ps_mi2)
```

Summary of balance for matched data:

	Means Treated	Means Control	Std. Mean Diff.	Var. Ratio	eCDF Mean
bdi1	30.96053	30.43421	0.05457687	1.0304654	0.02555668
cogn	82.73684	82.89474	-0.01523984	0.6461702	0.03562753
age	35.81579	36.17105	-0.05102740	0.8926150	0.02406015

	eCDF Max	Std. Pair Dist.
bdi1	0.11842105	0.5266668
cogn	0.14473684	1.0667890
age	0.06578947	1.1509514

```
plot(ps_mi2, type = "QQ",  
     which.xs = c("bdi1", "cogn", "age")) # Auswahl Kovariaten
```

