

# Introduction to cohort analysis

CUSTOMER SEGMENTATION IN PYTHON



**Karolis Urbonas**

Head of Data Science, Amazon

# About me



- Head of Data Science at Amazon
- 10+ years experience with analytics and ML
- Worked in:
  - eCommerce
  - banking
  - consulting
  - finance
  - other industries

# Prerequisites

- `pandas` library
- `datetime` objects
- basic plotting with `matplotlib` or `seaborn`
- basic knowledge of k-means clustering

# What is cohort analysis?

- Mutually exclusive segments - cohorts
- Compare metrics across **product** lifecycle
- Compare metrics across **customer** lifecycle

# Types of cohorts

- Time cohorts
- Behavior cohorts
- Size cohorts

# Elements of cohort analysis

- Pivot table

CohortIndex	1	2	3	4	5	6	7	8	9	10	11	12	13
CohortMonth													
2010-12-01	716.0	246.0	221.0	251.0	245.0	285.0	249.0	236.0	240.0	265.0	254.0	348.0	172.0
2011-01-01	332.0	69.0	82.0	81.0	110.0	90.0	82.0	86.0	104.0	102.0	124.0	45.0	NaN
2011-02-01	316.0	58.0	57.0	83.0	85.0	74.0	80.0	83.0	86.0	95.0	28.0	NaN	NaN
2011-03-01	388.0	63.0	100.0	76.0	83.0	67.0	98.0	85.0	107.0	38.0	NaN	NaN	NaN
2011-04-01	255.0	49.0	52.0	49.0	47.0	52.0	56.0	59.0	17.0	NaN	NaN	NaN	NaN
2011-05-01	249.0	40.0	43.0	36.0	52.0	58.0	61.0	22.0	NaN	NaN	NaN	NaN	NaN
2011-06-01	207.0	33.0	26.0	41.0	49.0	62.0	19.0	NaN	NaN	NaN	NaN	NaN	NaN
2011-07-01	173.0	28.0	31.0	38.0	44.0	17.0	NaN	NaN	NaN	NaN	NaN	NaN	NaN
2011-08-01	139.0	30.0	28.0	35.0	14.0	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
2011-09-01	279.0	56.0	78.0	34.0	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
2011-10-01	318.0	67.0	30.0	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
2011-11-01	291.0	32.0	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
2011-12-01	38.0	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN

# Elements of cohort analysis

- Pivot table
- Assigned cohort in rows

CohortIndex	1	2	3	4	5	6	7	8	9	10	11	12	13
CohortMonth													
2010-12-01	716.0	246.0	221.0	251.0	245.0	285.0	249.0	236.0	240.0	265.0	254.0	348.0	172.0
2011-01-01	332.0	69.0	82.0	81.0	110.0	90.0	82.0	86.0	104.0	102.0	124.0	45.0	NaN
2011-02-01	316.0	58.0	57.0	83.0	85.0	74.0	80.0	83.0	86.0	95.0	28.0	NaN	NaN
2011-03-01	388.0	63.0	100.0	76.0	83.0	67.0	98.0	85.0	107.0	38.0	NaN	NaN	NaN
2011-04-01	255.0	49.0	52.0	49.0	47.0	52.0	56.0	59.0	17.0	NaN	NaN	NaN	NaN
2011-05-01	249.0	40.0	43.0	36.0	52.0	58.0	61.0	22.0	NaN	NaN	NaN	NaN	NaN
2011-06-01	207.0	33.0	26.0	41.0	49.0	62.0	19.0	NaN	NaN	NaN	NaN	NaN	NaN
2011-07-01	173.0	28.0	31.0	38.0	44.0	17.0	NaN	NaN	NaN	NaN	NaN	NaN	NaN
2011-08-01	139.0	30.0	28.0	35.0	14.0	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
2011-09-01	279.0	56.0	78.0	34.0	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
2011-10-01	318.0	67.0	30.0	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
2011-11-01	291.0	32.0	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
2011-12-01	38.0	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN

# Elements of cohort analysis

- Pivot table
- Assigned cohort in **rows**
- Cohort Index in **columns**

CohortIndex	1	2	3	4	5	6	7	8	9	10	11	12	13
CohortMonth													
2010-12-01	716.0	246.0	221.0	251.0	245.0	285.0	249.0	236.0	240.0	265.0	254.0	348.0	172.0
2011-01-01	332.0	69.0	82.0	81.0	110.0	90.0	82.0	86.0	104.0	102.0	124.0	45.0	NaN
2011-02-01	316.0	58.0	57.0	83.0	85.0	74.0	80.0	83.0	86.0	95.0	28.0	NaN	NaN
2011-03-01	388.0	63.0	100.0	76.0	83.0	67.0	98.0	85.0	107.0	38.0	NaN	NaN	NaN
2011-04-01	255.0	49.0	52.0	49.0	47.0	52.0	56.0	59.0	17.0	NaN	NaN	NaN	NaN
2011-05-01	249.0	40.0	43.0	36.0	52.0	58.0	61.0	22.0	NaN	NaN	NaN	NaN	NaN
2011-06-01	207.0	33.0	26.0	41.0	49.0	62.0	19.0	NaN	NaN	NaN	NaN	NaN	NaN
2011-07-01	173.0	28.0	31.0	38.0	44.0	17.0	NaN	NaN	NaN	NaN	NaN	NaN	NaN
2011-08-01	139.0	30.0	28.0	35.0	14.0	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
2011-09-01	279.0	56.0	78.0	34.0	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
2011-10-01	318.0	67.0	30.0	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
2011-11-01	291.0	32.0	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
2011-12-01	38.0	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN



# Elements of cohort analysis

- Pivot table
- Assigned cohort in **rows**
- Cohort Index in **columns**
- Metrics in the **table**

CohortIndex	1	2	3	4	5	6	7	8	9	10	11	12	13
CohortMonth													
2010-12-01	716.0	246.0	221.0	251.0	245.0	285.0	249.0	236.0	240.0	265.0	254.0	348.0	172.0
2011-01-01	332.0	69.0	82.0	81.0	110.0	90.0	82.0	86.0	104.0	102.0	124.0	45.0	NaN
2011-02-01	316.0	58.0	57.0	83.0	85.0	74.0	80.0	83.0	86.0	95.0	28.0	NaN	NaN
2011-03-01	388.0	63.0	100.0	76.0	83.0	67.0	98.0	85.0	107.0	38.0	NaN	NaN	NaN
2011-04-01	255.0	49.0	52.0	49.0	47.0	52.0	56.0	59.0	17.0	NaN	NaN	NaN	NaN
2011-05-01	249.0	40.0	43.0	36.0	52.0	58.0	61.0	22.0	NaN	NaN	NaN	NaN	NaN
2011-06-01	207.0	33.0	26.0	41.0	49.0	62.0	19.0	NaN	NaN	NaN	NaN	NaN	NaN
2011-07-01	173.0	28.0	31.0	38.0	44.0	17.0	NaN	NaN	NaN	NaN	NaN	NaN	NaN
2011-08-01	139.0	30.0	28.0	35.0	14.0	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
2011-09-01	279.0	56.0	78.0	34.0	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
2011-10-01	318.0	67.0	30.0	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
2011-11-01	291.0	32.0	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
2011-12-01	38.0	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN

# Elements of cohort analysis

- First cohort was acquired in December 2010

CohortIndex	1	2	3	4	5	6	7	8	9	10	11	12	13
CohortMonth													
2010-12-01	716.0	246.0	221.0	251.0	245.0	285.0	249.0	236.0	240.0	265.0	254.0	348.0	172.0
2011-01-01	332.0	69.0	82.0	81.0	110.0	90.0	82.0	86.0	104.0	102.0	124.0	45.0	NaN
2011-02-01	316.0	58.0	57.0	83.0	85.0	74.0	80.0	83.0	86.0	95.0	28.0	NaN	NaN
2011-03-01	388.0	63.0	100.0	76.0	83.0	67.0	98.0	85.0	107.0	38.0	NaN	NaN	NaN
2011-04-01	255.0	49.0	52.0	49.0	47.0	52.0	56.0	59.0	17.0	NaN	NaN	NaN	NaN
2011-05-01	249.0	40.0	43.0	36.0	52.0	58.0	61.0	22.0	NaN	NaN	NaN	NaN	NaN
2011-06-01	207.0	33.0	26.0	41.0	49.0	62.0	19.0	NaN	NaN	NaN	NaN	NaN	NaN
2011-07-01	173.0	28.0	31.0	38.0	44.0	17.0	NaN	NaN	NaN	NaN	NaN	NaN	NaN
2011-08-01	139.0	30.0	28.0	35.0	14.0	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
2011-09-01	279.0	56.0	78.0	34.0	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
2011-10-01	318.0	67.0	30.0	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
2011-11-01	291.0	32.0	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
2011-12-01	38.0	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN

# Elements of cohort analysis

- First cohort was acquired in December 2010
- Last cohort was acquired in December 2011

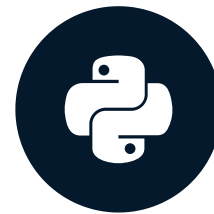
CohortIndex	1	2	3	4	5	6	7	8	9	10	11	12	13
CohortMonth													
2010-12-01	716.0	246.0	221.0	251.0	245.0	285.0	249.0	236.0	240.0	265.0	254.0	348.0	172.0
2011-01-01	332.0	69.0	82.0	81.0	110.0	90.0	82.0	86.0	104.0	102.0	124.0	45.0	NaN
2011-02-01	316.0	58.0	57.0	83.0	85.0	74.0	80.0	83.0	86.0	95.0	28.0	NaN	NaN
2011-03-01	388.0	63.0	100.0	76.0	83.0	67.0	98.0	85.0	107.0	38.0	NaN	NaN	NaN
2011-04-01	255.0	49.0	52.0	49.0	47.0	52.0	56.0	59.0	17.0	NaN	NaN	NaN	NaN
2011-05-01	249.0	40.0	43.0	36.0	52.0	58.0	61.0	22.0	NaN	NaN	NaN	NaN	NaN
2011-06-01	207.0	33.0	26.0	41.0	49.0	62.0	19.0	NaN	NaN	NaN	NaN	NaN	NaN
2011-07-01	173.0	28.0	31.0	38.0	44.0	17.0	NaN	NaN	NaN	NaN	NaN	NaN	NaN
2011-08-01	139.0	30.0	28.0	35.0	14.0	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
2011-09-01	279.0	56.0	78.0	34.0	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
2011-10-01	318.0	67.0	30.0	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
2011-11-01	291.0	32.0	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
2011-12-01	38.0	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN

# Explore the cohort table

CUSTOMER SEGMENTATION IN PYTHON

# Cohort analysis

## CUSTOMER SEGMENTATION IN PYTHON



**Karolis Urbonas**

Head of Data Science, Amazon

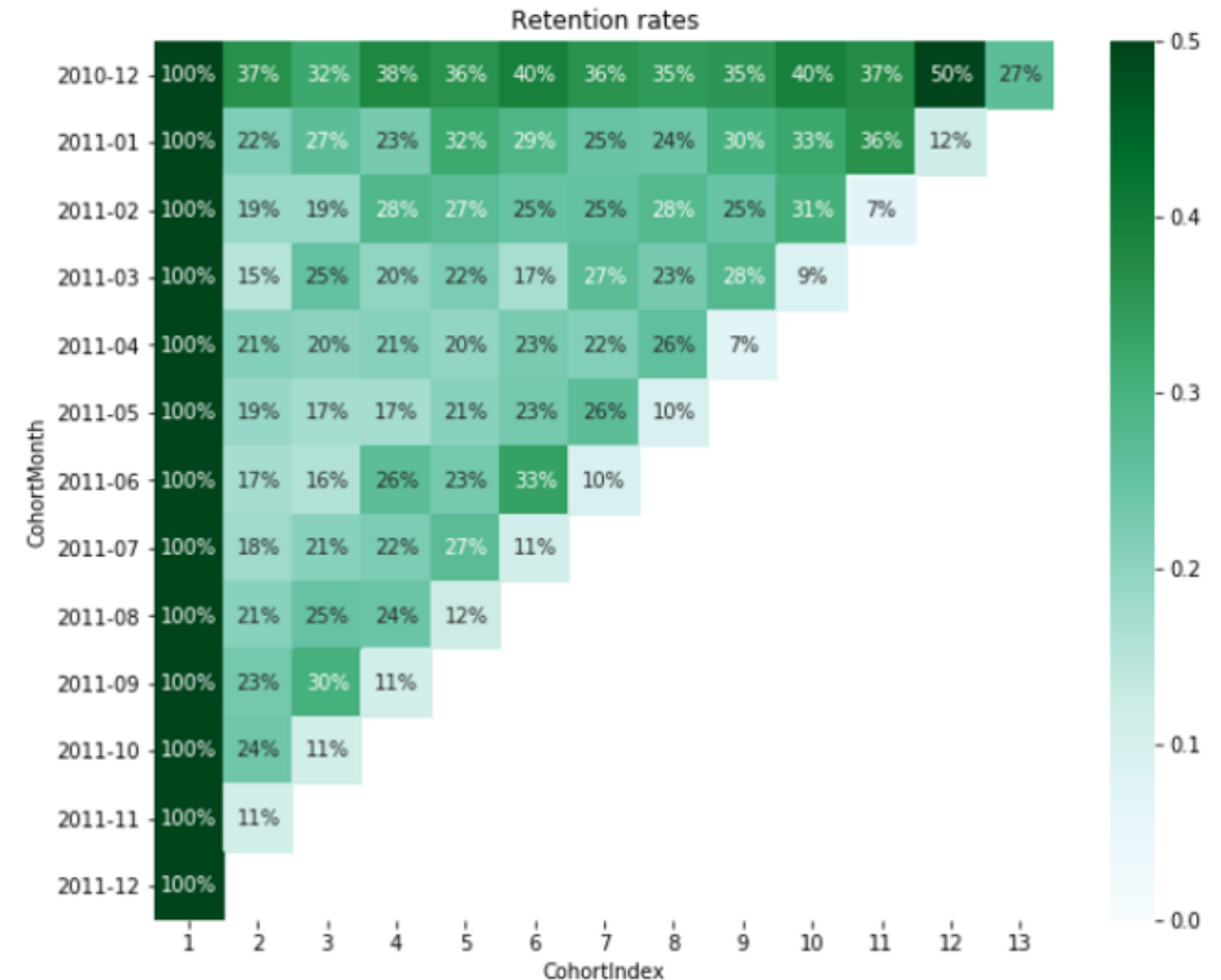
# Cohort analysis heatmap

Rows:

- First activity
- Here - month of acquisition

Columns:

- Time since first activity
- Here - months since acquisition



# Cohort analysis heatmap

Rows:

- First activity
- Here - month of acquisition

Columns:

- Time since first activity
- Here - months since acquisition



# Online retail data

Over 0.5 million transactions from a UK-based online retail store.

We will use a randomly sampled 20% subset of this dataset throughout the course.



## Online Retail Data Set

*Download:* [Data Folder](#), [Data Set Description](#)



# Top 5 rows of data

```
online.head()
```

InvoiceNo	StockCode	Description	Quantity	InvoiceDate	UnitPrice	CustomerID	Country
572558	22745	POPPY'S PLAYHOUSE BEDROOM	6	2011-10-25 08:26:00	2.10	14286	United Kingdom
577485	23196	VINTAGE LEAF MAGNETIC NOTEPAD	1	2011-11-20 11:56:00	1.45	16360	United Kingdom
560034	23299	FOOD COVER WITH BEADS SET 2	6	2011-07-14 13:35:00	3.75	13933	United Kingdom
578307	72349B	SET/6 PURPLE BUTTERFLY T-LIGHTS	1	2011-11-23 15:53:00	2.10	17290	United Kingdom
554656	21756	BATH BUILDING BLOCK WORD	3	2011-05-25 13:36:00	5.95	17663	United Kingdom

# Assign acquisition month cohort

```
def get_month(x): return dt.datetime(x.year, x.month, 1)
online['InvoiceMonth'] = online['InvoiceDate'].apply(get_month)
grouping = online.groupby('CustomerID')['InvoiceMonth']
online['CohortMonth'] = grouping.transform('min')
online.head()
```

InvoiceNo	StockCode	Description	Quantity	InvoiceDate	UnitPrice	CustomerID	Country	InvoiceMonth	CohortMonth
416792	572558	22745 POPPY'S PLAYHOUSE BEDROOM	6	2011-10-25 08:26:00	2.10	14286.0	United Kingdom	2011-10-01	2011-04-01
482904	577485	23196 VINTAGE LEAF MAGNETIC NOTEPAD	1	2011-11-20 11:56:00	1.45	16360.0	United Kingdom	2011-11-01	2011-09-01
263743	560034	23299 FOOD COVER WITH BEADS SET 2	6	2011-07-14 13:35:00	3.75	13933.0	United Kingdom	2011-07-01	2011-07-01
495549	578307	72349B SET/6 PURPLE BUTTERFLY T-LIGHTS	1	2011-11-23 15:53:00	2.10	17290.0	United Kingdom	2011-11-01	2011-11-01
204384	554656	21756 BATH BUILDING BLOCK WORD	3	2011-05-25 13:36:00	5.95	17663.0	United Kingdom	2011-05-01	2011-02-01

# Extract integer values from data

Define function to extract `year`, `month` and `day` integer values.

We will use it throughout the course.

```
def get_date_int(df, column):  
    year = df[column].dt.year  
    month = df[column].dt.month  
    day = df[column].dt.day  
    return year, month, day
```

# Assign time offset value

```
invoice_year, invoice_month, _ = get_date_int(online, 'InvoiceMonth')
cohort_year, cohort_month, _ = get_date_int(online, 'CohortMonth')
years_diff = invoice_year - cohort_year
months_diff = invoice_month - cohort_month
online['CohortIndex'] = years_diff * 12 + months_diff + 1
online.head()
```

InvoiceNo	StockCode	Description	Quantity	InvoiceDate	UnitPrice	CustomerID	Country	InvoiceMonth	CohortMonth	CohortIndex
416792	572558	POPPY'S PLAYHOUSE BEDROOM	6	2011-10-25 08:26:00	2.10	14286.0	United Kingdom	2011-10-01	2011-04-01	7
482904	577485	VINTAGE LEAF MAGNETIC NOTEPAD	1	2011-11-20 11:56:00	1.45	16360.0	United Kingdom	2011-11-01	2011-09-01	3
263743	560034	FOOD COVER WITH BEADS SET 2	6	2011-07-14 13:35:00	3.75	13933.0	United Kingdom	2011-07-01	2011-07-01	1
495549	578307	SET/6 PURPLE BUTTERFLY T-LIGHTS	1	2011-11-23 15:53:00	2.10	17290.0	United Kingdom	2011-11-01	2011-11-01	1
204384	554656	BATH BUILDING BLOCK WORD	3	2011-05-25 13:36:00	5.95	17663.0	United Kingdom	2011-05-01	2011-02-01	4

# Count monthly active customers from each cohort

```
grouping = online.groupby(['CohortMonth', 'CohortIndex'])
cohort_data = grouping['CustomerID'].apply(pd.Series.nunique)
cohort_data = cohort_data.reset_index()
cohort_counts = cohort_data.pivot(index='CohortMonth',
                                   columns='CohortIndex',
                                   values='CustomerID')

print(cohort_counts)
```

CohortIndex	1	2	3	4	5	6	7	8	9	10	11	12	13
CohortMonth													
2010-12-01	716.0	246.0	221.0	251.0	245.0	285.0	249.0	236.0	240.0	265.0	254.0	348.0	172.0
2011-01-01	332.0	69.0	82.0	81.0	110.0	90.0	82.0	86.0	104.0	102.0	124.0	45.0	NaN
2011-02-01	316.0	58.0	57.0	83.0	85.0	74.0	80.0	83.0	86.0	95.0	28.0	NaN	NaN
2011-03-01	388.0	63.0	100.0	76.0	83.0	67.0	98.0	85.0	107.0	38.0	NaN	NaN	NaN
2011-04-01	255.0	49.0	52.0	49.0	47.0	52.0	56.0	59.0	17.0	NaN	NaN	NaN	NaN
2011-05-01	249.0	40.0	43.0	36.0	52.0	58.0	61.0	22.0	NaN	NaN	NaN	NaN	NaN
2011-06-01	207.0	33.0	26.0	41.0	49.0	62.0	19.0	NaN	NaN	NaN	NaN	NaN	NaN
2011-07-01	173.0	28.0	31.0	38.0	44.0	17.0	NaN	NaN	NaN	NaN	NaN	NaN	NaN
2011-08-01	139.0	30.0	28.0	35.0	14.0	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
2011-09-01	279.0	56.0	78.0	34.0	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
2011-10-01	318.0	67.0	30.0	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
2011-11-01	291.0	32.0	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
2011-12-01	38.0	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN

# Your turn to build some cohorts!

CUSTOMER SEGMENTATION IN PYTHON

# Cohort metrics

CUSTOMER SEGMENTATION IN PYTHON



**Karolis Urbonas**

Head of Data Science, Amazon



# Customer retention: cohort\_counts table

- How many customers originally in each cohort in the cohort\_counts table?

CohortIndex	1	2	3	4	5	6	7	8	9	10	11	12	13
CohortMonth													
2010-12-01	716.0	246.0	221.0	251.0	245.0	285.0	249.0	236.0	240.0	265.0	254.0	348.0	172.0
2011-01-01	332.0	69.0	82.0	81.0	110.0	90.0	82.0	86.0	104.0	102.0	124.0	45.0	NaN
2011-02-01	316.0	58.0	57.0	83.0	85.0	74.0	80.0	83.0	86.0	95.0	28.0	NaN	NaN
2011-03-01	388.0	63.0	100.0	76.0	83.0	67.0	98.0	85.0	107.0	38.0	NaN	NaN	NaN
2011-04-01	255.0	49.0	52.0	49.0	47.0	52.0	56.0	59.0	17.0	NaN	NaN	NaN	NaN
2011-05-01	249.0	40.0	43.0	36.0	52.0	58.0	61.0	22.0	NaN	NaN	NaN	NaN	NaN
2011-06-01	207.0	33.0	26.0	41.0	49.0	62.0	19.0	NaN	NaN	NaN	NaN	NaN	NaN
2011-07-01	173.0	28.0	31.0	38.0	44.0	17.0	NaN	NaN	NaN	NaN	NaN	NaN	NaN
2011-08-01	139.0	30.0	28.0	35.0	14.0	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
2011-09-01	279.0	56.0	78.0	34.0	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
2011-10-01	318.0	67.0	30.0	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
2011-11-01	291.0	32.0	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
2011-12-01	38.0	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN

# Customer retention: cohort\_counts table

- How many customers originally in each cohort?
- How many of them were active in following months?

CohortIndex	1	2	3	4	5	6	7	8	9	10	11	12	13
CohortMonth													
2010-12-01	716.0	246.0	221.0	251.0	245.0	285.0	249.0	236.0	240.0	265.0	254.0	348.0	172.0
2011-01-01	332.0	69.0	82.0	81.0	110.0	90.0	82.0	86.0	104.0	102.0	124.0	45.0	NaN
2011-02-01	316.0	58.0	57.0	83.0	85.0	74.0	80.0	83.0	86.0	95.0	28.0	NaN	NaN
2011-03-01	388.0	63.0	100.0	76.0	83.0	67.0	98.0	85.0	107.0	38.0	NaN	NaN	NaN
2011-04-01	255.0	49.0	52.0	49.0	47.0	52.0	56.0	59.0	17.0	NaN	NaN	NaN	NaN
2011-05-01	249.0	40.0	43.0	36.0	52.0	58.0	61.0	22.0	NaN	NaN	NaN	NaN	NaN
2011-06-01	207.0	33.0	26.0	41.0	49.0	62.0	19.0	NaN	NaN	NaN	NaN	NaN	NaN
2011-07-01	173.0	28.0	31.0	38.0	44.0	17.0	NaN	NaN	NaN	NaN	NaN	NaN	NaN
2011-08-01	139.0	30.0	28.0	35.0	14.0	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
2011-09-01	279.0	56.0	78.0	34.0	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
2011-10-01	318.0	67.0	30.0	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
2011-11-01	291.0	32.0	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
2011-12-01	38.0	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN

# Calculate retention rate

1. Store the first column as `cohort_sizes`

```
cohort_sizes = cohort_counts.iloc[:,0]
```

2. Divide all values in the `cohort_counts` table by `cohort_sizes`

```
retention = cohort_counts.divide(cohort_sizes, axis=0)
```

3. Review the retention table

```
retention.round(3) * 100
```

# Retention table

CohortIndex	1	2	3	4	5	6	7	8	9	10	11	12	13
CohortMonth													
2010-12	100.0	34.4	30.9	35.1	34.2	39.8	34.8	33.0	33.5	37.0	35.5	48.6	24.0
2011-01	100.0	20.8	24.7	24.4	33.1	27.1	24.7	25.9	31.3	30.7	37.3	13.6	NaN
2011-02	100.0	18.4	18.0	26.3	26.9	23.4	25.3	26.3	27.2	30.1	8.9	NaN	NaN
2011-03	100.0	16.2	25.8	19.6	21.4	17.3	25.3	21.9	27.6	9.8	NaN	NaN	NaN
2011-04	100.0	19.2	20.4	19.2	18.4	20.4	22.0	23.1	6.7	NaN	NaN	NaN	NaN
2011-05	100.0	16.1	17.3	14.5	20.9	23.3	24.5	8.8	NaN	NaN	NaN	NaN	NaN
2011-06	100.0	15.9	12.6	19.8	23.7	30.0	9.2	NaN	NaN	NaN	NaN	NaN	NaN
2011-07	100.0	16.2	17.9	22.0	25.4	9.8	NaN	NaN	NaN	NaN	NaN	NaN	NaN
2011-08	100.0	21.6	20.1	25.2	10.1	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
2011-09	100.0	20.1	28.0	12.2	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
2011-10	100.0	21.1	9.4	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
2011-11	100.0	11.0	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
2011-12	100.0	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN

# Other metrics

```
grouping = online.groupby(['CohortMonth', 'CohortIndex'])
cohort_data = grouping['Quantity'].mean()
cohort_data = cohort_data.reset_index()
average_quantity = cohort_data.pivot(index='CohortMonth',
                                      columns='CohortIndex',
                                      values='Quantity')

average_quantity.round(1)
```

# Average quantity for each cohort

CohortIndex	1	2	3	4	5	6	7	8	9	10	11	12	13
CohortMonth													
2010-12	11.1	12.3	12.2	13.2	13.7	11.3	10.6	12.8	13.2	13.0	14.5	10.9	13.7
2011-01	10.9	10.8	10.0	10.1	14.3	13.2	17.4	16.4	18.7	10.2	10.7	13.2	NaN
2011-02	12.1	14.3	10.6	11.5	17.5	12.2	17.3	13.2	13.4	15.9	14.3	NaN	NaN
2011-03	9.6	14.2	13.0	10.2	16.1	12.7	11.6	11.5	9.0	9.6	NaN	NaN	NaN
2011-04	9.9	11.1	12.4	11.5	11.4	7.7	10.4	9.4	6.6	NaN	NaN	NaN	NaN
2011-05	14.1	9.6	15.3	11.6	11.9	8.5	9.8	7.3	NaN	NaN	NaN	NaN	NaN
2011-06	10.6	16.1	18.1	11.2	12.4	7.2	9.7	NaN	NaN	NaN	NaN	NaN	NaN
2011-07	10.9	16.4	5.6	10.1	6.2	7.0	NaN	NaN	NaN	NaN	NaN	NaN	NaN
2011-08	10.5	7.4	5.5	5.7	6.0	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
2011-09	11.5	6.3	8.4	9.9	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
2011-10	9.3	7.5	6.0	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
2011-11	7.8	7.1	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
2011-12	21.3	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN

# Let's practice on other cohort metrics!

CUSTOMER SEGMENTATION IN PYTHON

# Cohort analysis visualization

CUSTOMER SEGMENTATION IN PYTHON



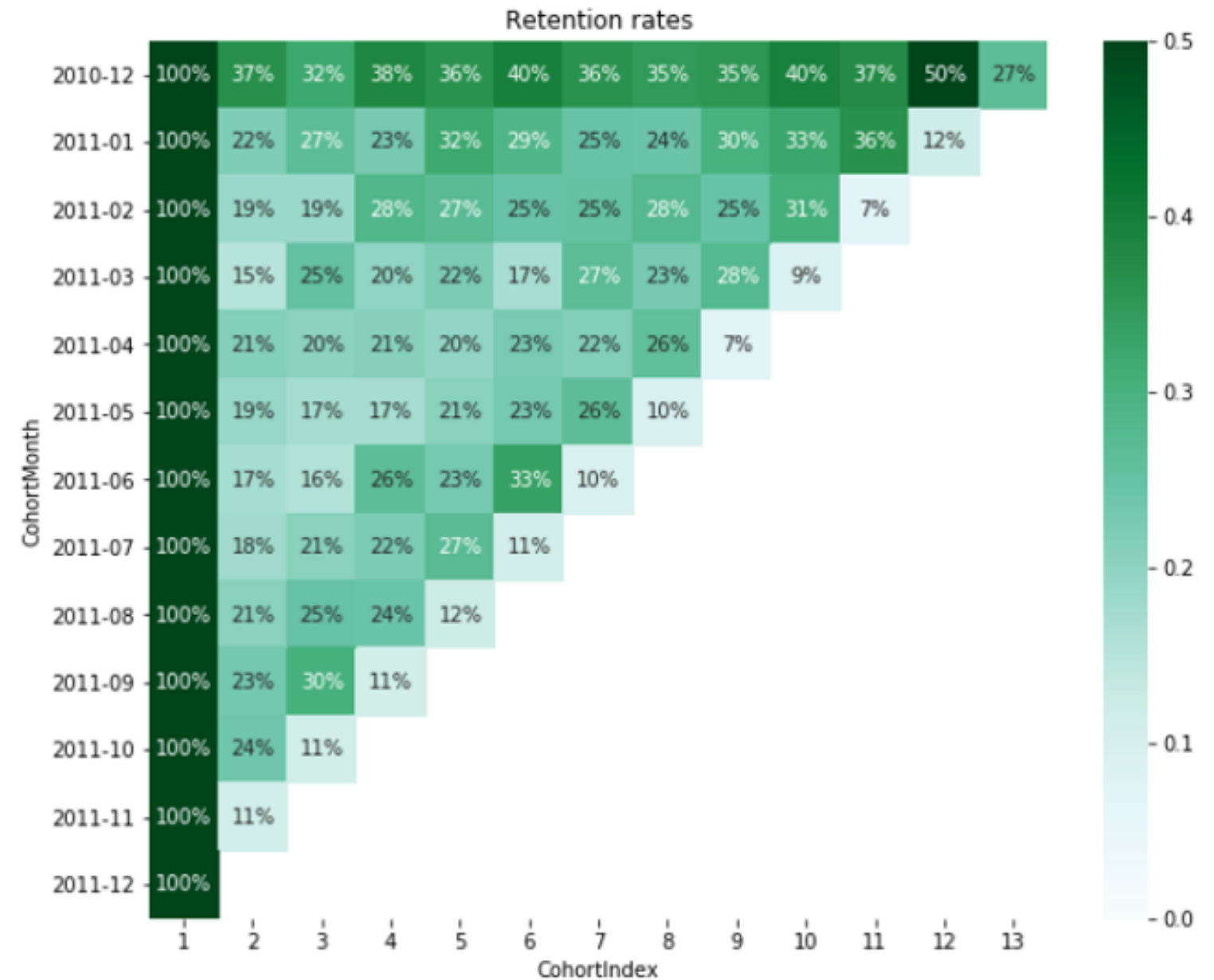
**Karolis Urbonas**

Head of Data Science, Amazon



# Heatmap

- Easiest way to visualize cohort analysis
- Includes both data and visuals
- Only few lines of code with `seaborn`



# Load the retention table

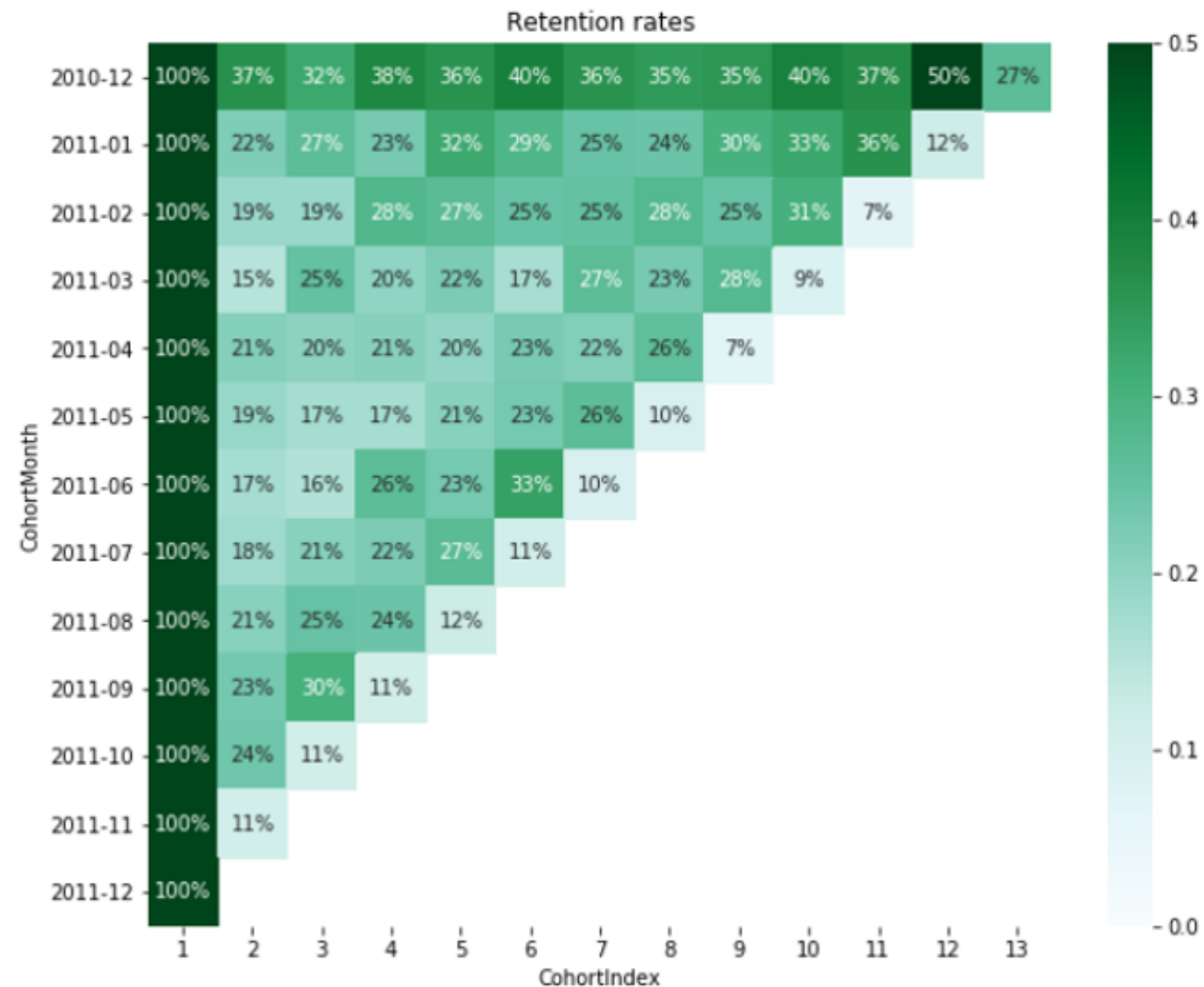
```
retention.round(3)*100
```

	1	2	3	4	5	6	7	8	9	10	11	12	13
CohortMonth													
2010-12	100.0	34.4	30.9	35.1	34.2	39.8	34.8	33.0	33.5	37.0	35.5	48.6	24.0
2011-01	100.0	20.8	24.7	24.4	33.1	27.1	24.7	25.9	31.3	30.7	37.3	13.6	NaN
2011-02	100.0	18.4	18.0	26.3	26.9	23.4	25.3	26.3	27.2	30.1	8.9	NaN	NaN
2011-03	100.0	16.2	25.8	19.6	21.4	17.3	25.3	21.9	27.6	9.8	NaN	NaN	NaN
2011-04	100.0	19.2	20.4	19.2	18.4	20.4	22.0	23.1	6.7	NaN	NaN	NaN	NaN
2011-05	100.0	16.1	17.3	14.5	20.9	23.3	24.5	8.8	NaN	NaN	NaN	NaN	NaN
2011-06	100.0	15.9	12.6	19.8	23.7	30.0	9.2	NaN	NaN	NaN	NaN	NaN	NaN
2011-07	100.0	16.2	17.9	22.0	25.4	9.8	NaN	NaN	NaN	NaN	NaN	NaN	NaN
2011-08	100.0	21.6	20.1	25.2	10.1	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
2011-09	100.0	20.1	28.0	12.2	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
2011-10	100.0	21.1	9.4	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
2011-11	100.0	11.0	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
2011-12	100.0	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN

# Build the heatmap

```
import seaborn as sns
import matplotlib.pyplot as plt
plt.figure(figsize=(10, 8))
plt.title('Retention rates')
sns.heatmap(data = retention,
            annot = True,
            fmt = '.0%',
            vmin = 0.0,
            vmax = 0.5,
            cmap = 'BuGn')
plt.show()
```

# Retention heatmap



# Practice visualizing cohorts

CUSTOMER SEGMENTATION IN PYTHON