Forecasting with cohort-based models

Nicolai Vicol
Data Scientist at Wix
June 3, 2021

About WIX

About WIX

Wix offers solutions to **create a professional website** and **manage your business**.

Whether you are an entrepreneur, designer, developer, musician, photographer or restaurant owner, Wix has the right product and features to fit your needs.

Wix products are provided on a **freemium** base.

Wix earns from **premium** upgrades, like domain, storage, bandwidth, online payments, etc.

Wix Premium Plans

Wix gives you 100s of templates, unlimited pages & top grade hosting FREE.
Upgrade to Premium and get even more.

Get Started Today

Website Plans
Great for showcasing a professional site

Business & eCommerce Plans
Essential for accepting online payments

	VIP	Unlimited (MOST POPULAR)	Combo	Connect Domain
	First Priority Support	Entrepreneurs & Freelancers	For Personal Use	Most Basic This Plan displays Wix ads
	€ 24 50 /month	€ 12 50 /month	€ 8 50 /month	€ 4 50 /month
Custom Domain	✓	✓	✓	✓
Free Domain for 1 Year	✓	✓	✓	–
Remove Wix Ads	✓	✓	✓	–
Free SSL Certificate	✓	✓	✓	✓
Bandwidth	Unlimited	Unlimited	2GB	1GB
Storage Space	35GB	10GB	3GB	500MB

200 M

Registered Users

>5.5 M

Premium Subscriptions

WiX

** as of 2020-12-31, from investor presentation, https://investors.wix.com/*

The ideas exposed here are **applicable** to **any company** offering a **freemium** product or a **free trial** of it, which is **free** in a basic or limited setup, but can be **upgraded** with **premium subscriptions**, for example Spotify, Dropbox, Slack, Mailchimp.

Spotify

Pick your Premium

Listen without limits on your phone, speaker, and other devices.

1 month free

Individual

$9.99/month after offer period
1 account

- Listen to music ad-free
- Play anywhere - even offline
- On-demand playback

GET STARTED

Terms and conditions apply. 1 month free not available for users who have already tried Premium.

1 month free

Duo

$12.99/month after offer period
2 accounts

- 2 Premium accounts for a couple under one roof
- Duo Mix: a playlist for two, regularly updated with music you both enjoy
- Ad-free music listening, play offline, on-demand playback

GET STARTED

Terms and conditions apply. 1 month free not available for users who have already tried Premium.

1 month free

Family

$15.99/month after offer period
6 accounts

- 6 Premium accounts for family members living under one roof
- Family Mix: a playlist for your family, regularly updated with music you all enjoy
- Block explicit music
- Ad-free music listening, play offline, on-demand playback
- Spotify Kids: a separate app made just for kids

GET STARTED

Terms and conditions apply. 1 month free not available for users who have already tried Premium.

1 month free

Student

$4.99/month after offer period
1 account

- Hulu (ad-supported) plan
- SHOWTIME
- Listen to music ad-free
- Play anywhere - even offline
- On-demand playback

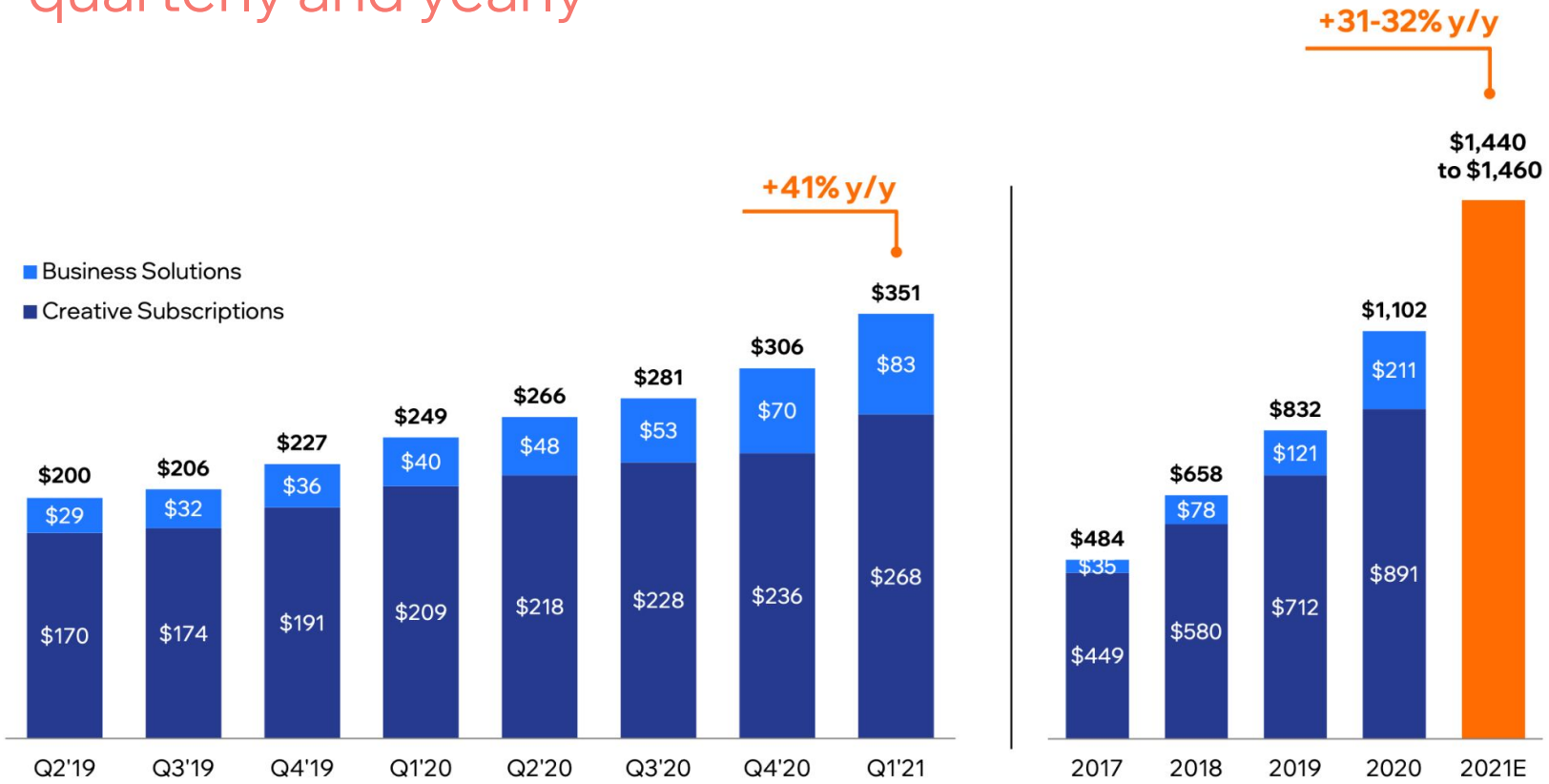
GET STARTED

Offer currently includes access to Hulu (ad-supported) plan and SHOWTIME Streaming Service, subject to eligibility. Available only to students at an accredited higher education institution. 1 month free only open to higher education students who haven't already tried Premium. Terms and conditions apply.

Dropbox

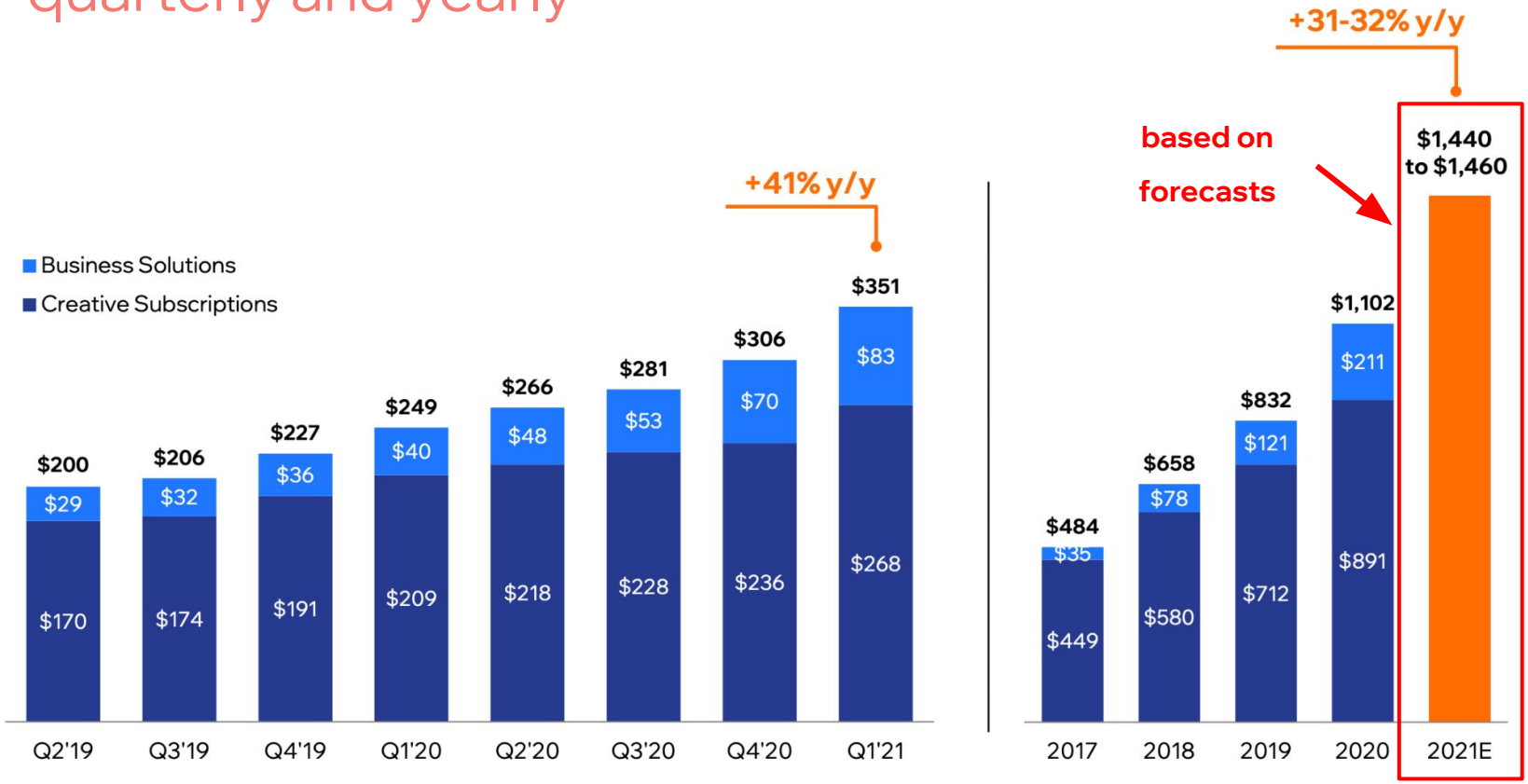
	Personal		Business		
	Plus For individuals Buy now	**Family** For families Buy now	**Professional** For individuals Try for free or purchase now	**Standard** For smaller teams Try for free or purchase now	**Advanced** For larger teams Try for free or purchase now
Dropbox core features					
Storage	2 TB (2,000 GB)	Share 2 TB (2,000 GB)	3 TB (3,000 GB)	5 TB (5,000 GB)	As much space as needed
Users	1 user	Up to 6 users	1 user	3+ users	3+ users
Best-in-class sync technology	✓	✓	✓	✓	✓
Anytime, anywhere access	✓	✓	✓	✓	✓
Computer backup	✓	✓	✓	✓	✓
Easy and secure sharing	✓	✓	✓	✓	✓
256-bit AES and SSL/TLS encryption	✓	✓	✓	✓	✓
Billed yearly Billed monthly	For individuals $9.99 / month Buy now	For families $16.99 / month Buy now	For individuals $16.58 / month Try for free or purchase now	For smaller teams $12.50 / user / month Try for free or purchase now	For larger teams $20 / user / month Try for free or purchase now

Revenue of Wix
quarterly and yearly



** as of 2021-03-31, from investor presentation, find more: https://investors.wix.com/*

Revenue of Wix
quarterly and yearly

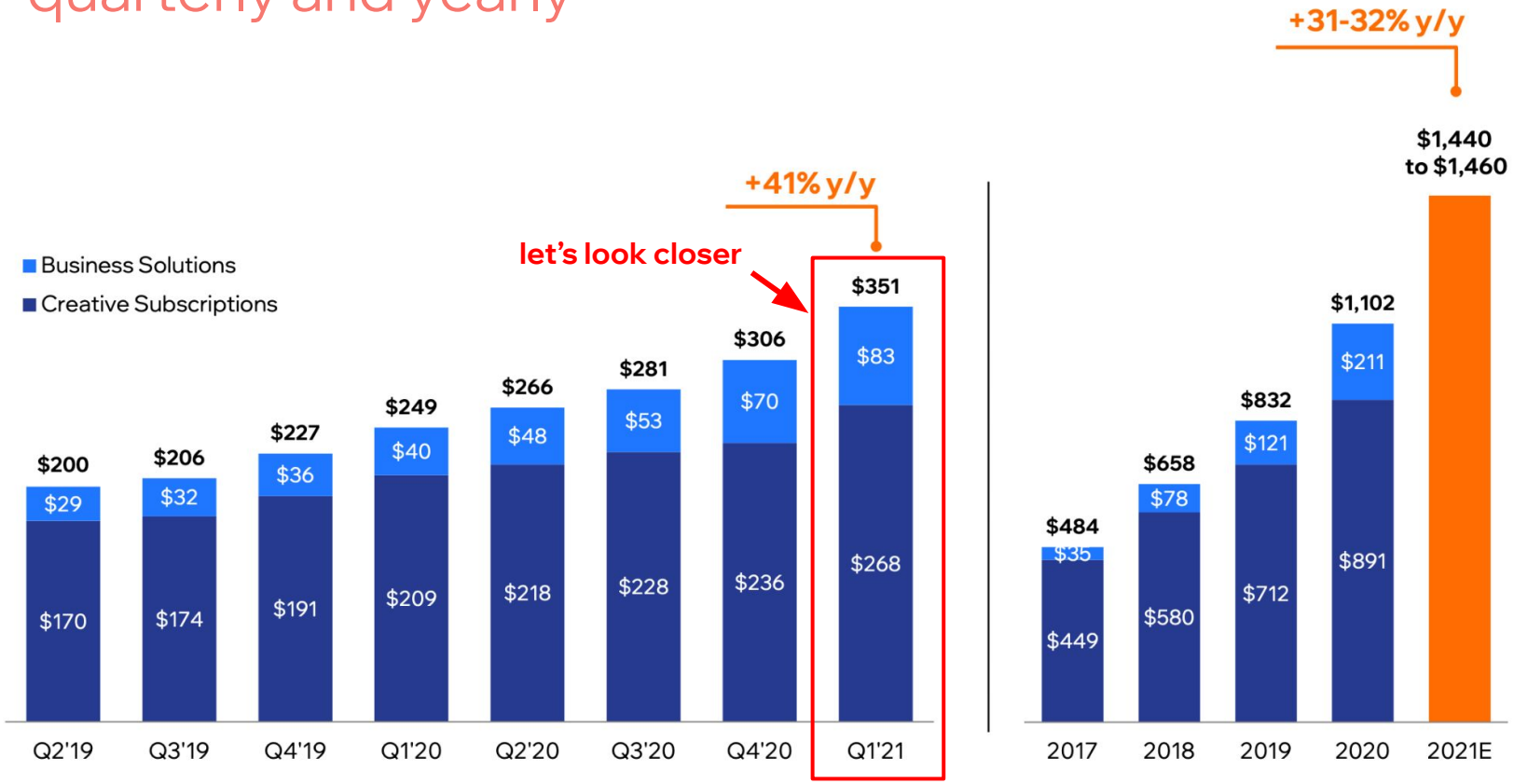


** as of 2021-03-31, from investor presentation, find more: https://investors.wix.com/*

Our task as forecasters:

Forecast future incoming cash flows for **operational** and **strategic** planning by **management**, and also support the guidance provided to **investors**.

Revenue of Wix
quarterly and yearly



** as of 2021-03-31, from investor presentation, find more: https://investors.wix.com/*

Breakdown by main categories

Business Solutions & Creative Subscriptions

Business Solutions

Google Workspace,
Ascend, Ads, TPAs

Creative Subscriptions

Type: Website,
Business/eCommerce
Term: Monthly, 1, 2 & 3 Years
Geo: US, non-US

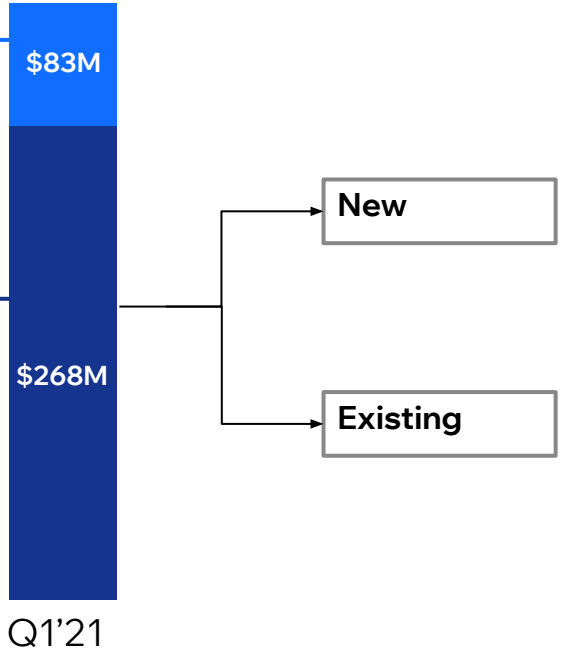
Breakdown of existing
Renewed or switched to other plans

Business Solutions

Google Workspace,
Ascend, Ads, TPAs

Creative Subscriptions

Type: Website,
Business/eCommerce
Term: Monthly, 1, 2 & 3 Years
Geo: US, non-US



Breakdown of existing
Renewed or switched to other plans

Business Solutions

Google Workspace,

Ascend, Ads, TPAs

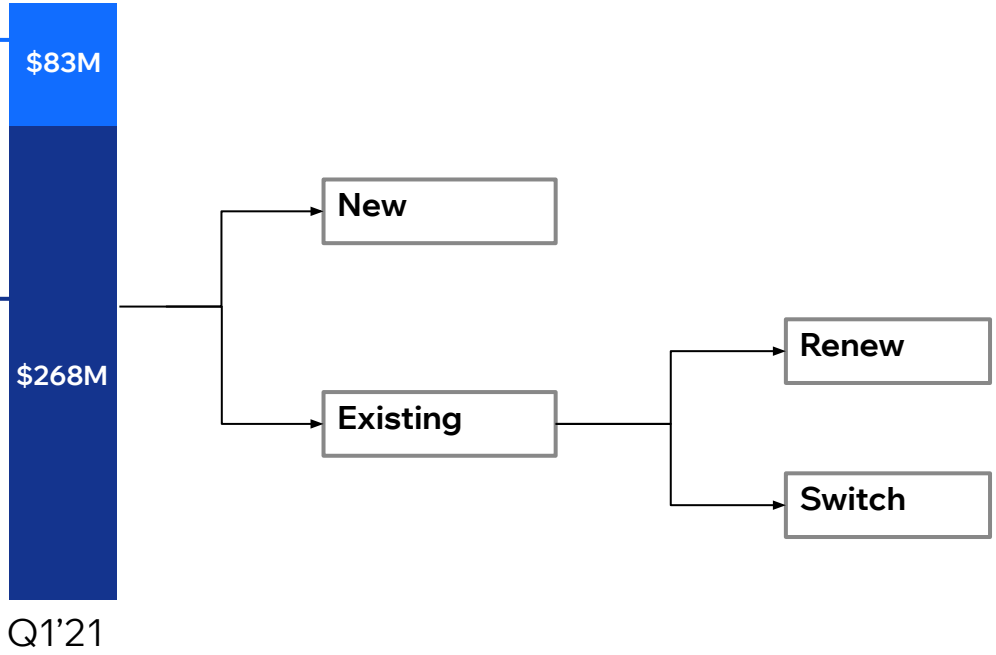
Creative Subscriptions

Type: Website,

Business/eCommerce

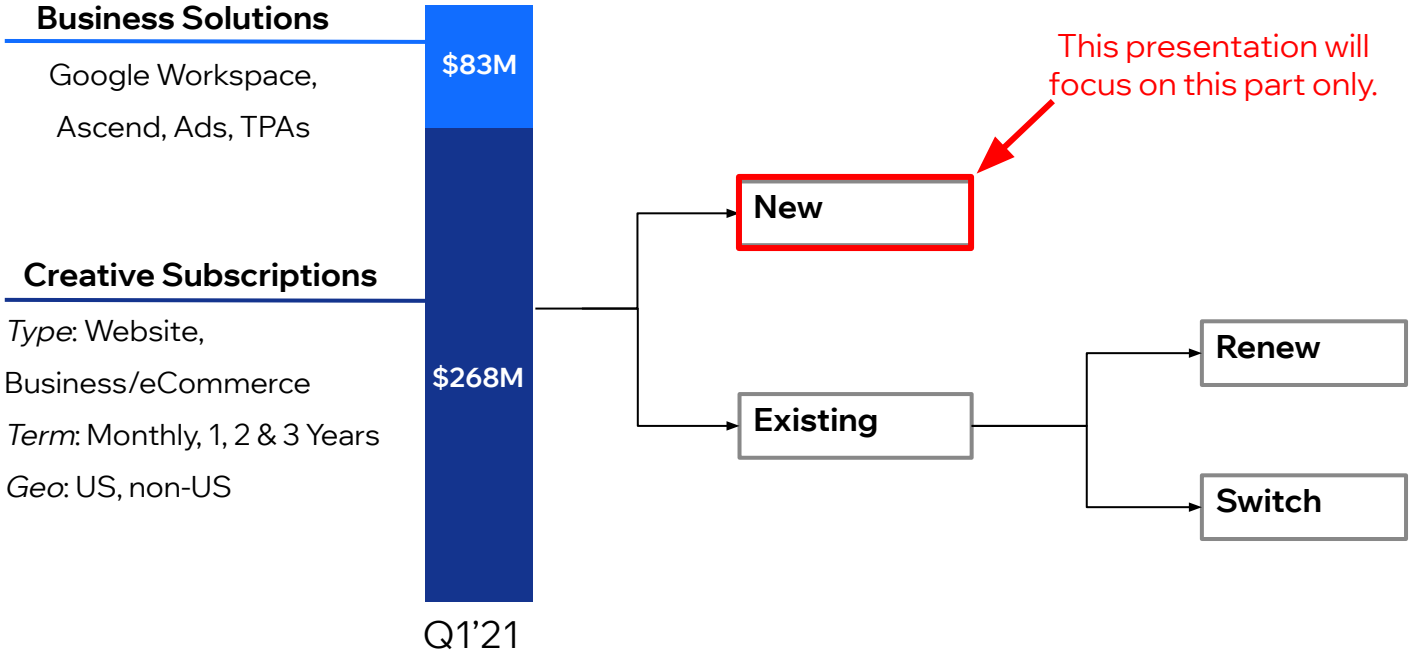
Term: Monthly, 1, 2 & 3 Years

Geo: US, non-US



Q1'21

Breakdown of existing
Renewed or switched to other plans



Cohorts of users

What is a cohort?

cohort

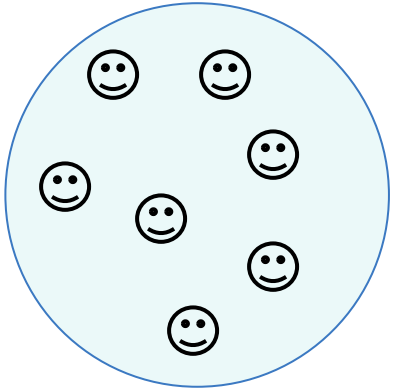
noun [C, + sing/pl verb]

UK /ˈkəʊ.hɔːt/ **US** /ˈkoʊ.hɔːrt/

SOCIAL SCIENCE • specialized

a group of people who share a characteristic, usually age:

- *This study followed up a cohort of 386 patients aged 65+ for six months after their discharge home.*



Cohorts at Wix:

users who registered within the same time period

examples

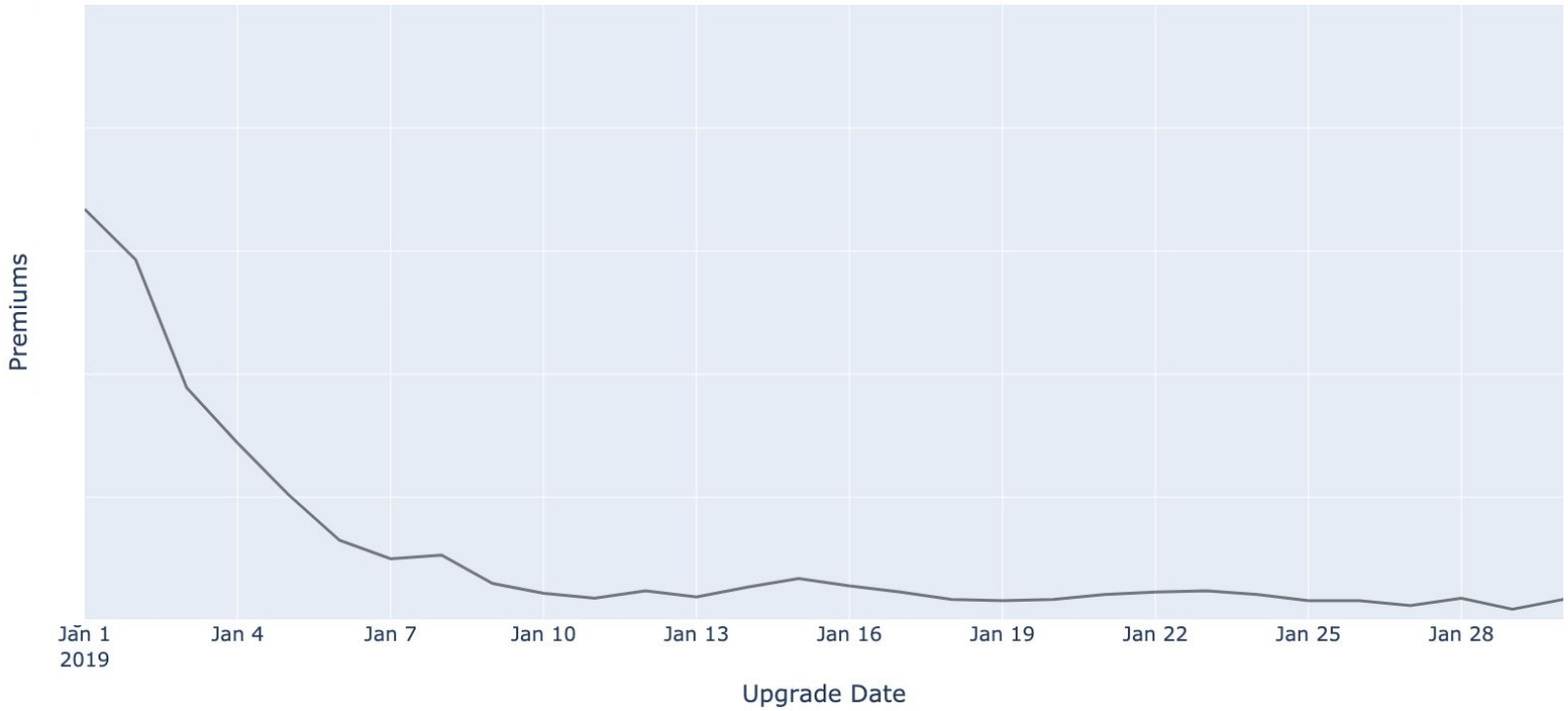
Cohorts behaviour and properties

Disclaimer:

This presentation **does not use Wix's actual data**.
All charts and numbers in what follows are based on **synthetic data**, generated to **mimic** behaviours and properties that are likely to be observed for users of **any product company** with **freemium** business model.

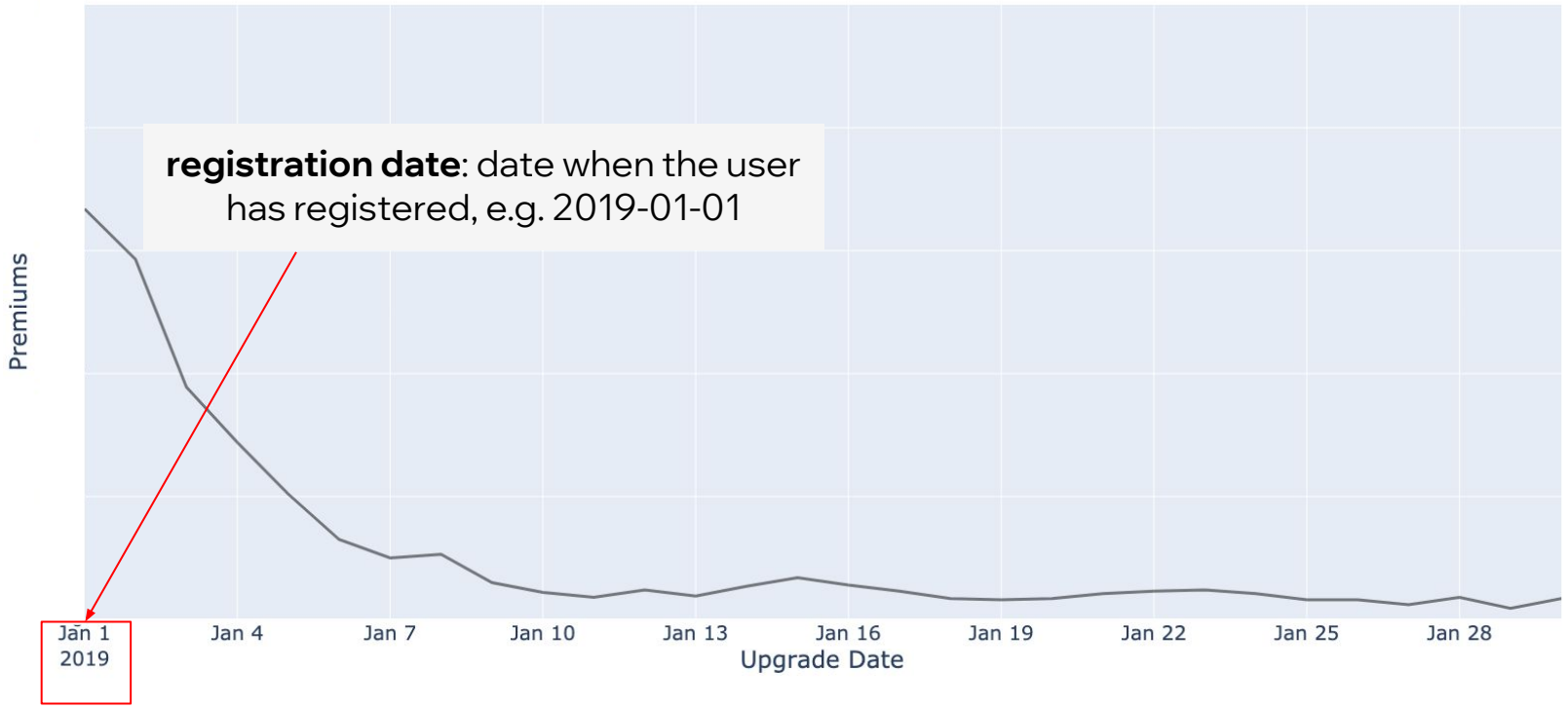
Premiums by upgrade date

for the cohort of registration date: 2019-01-01



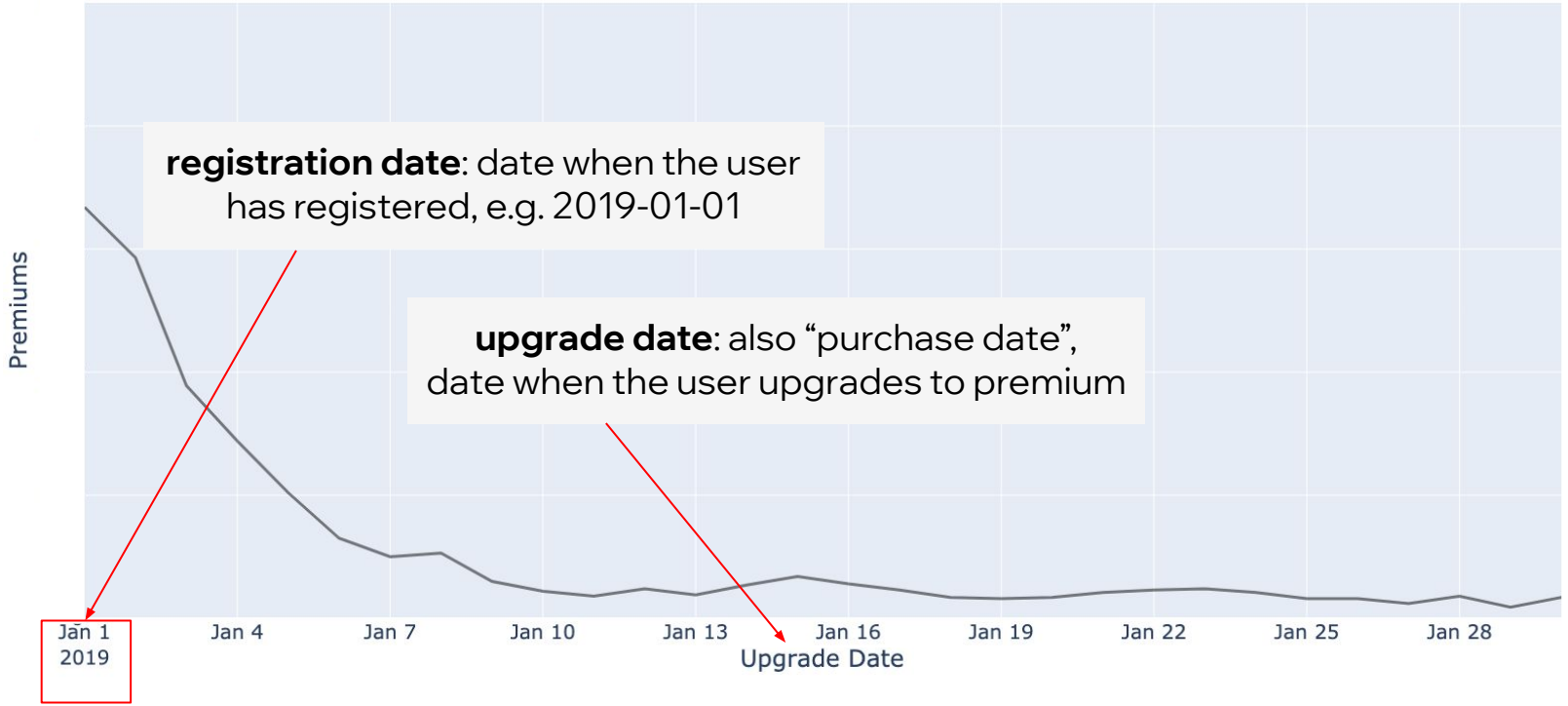
Premiums by upgrade date

for the cohort of registration date: 2019-01-01



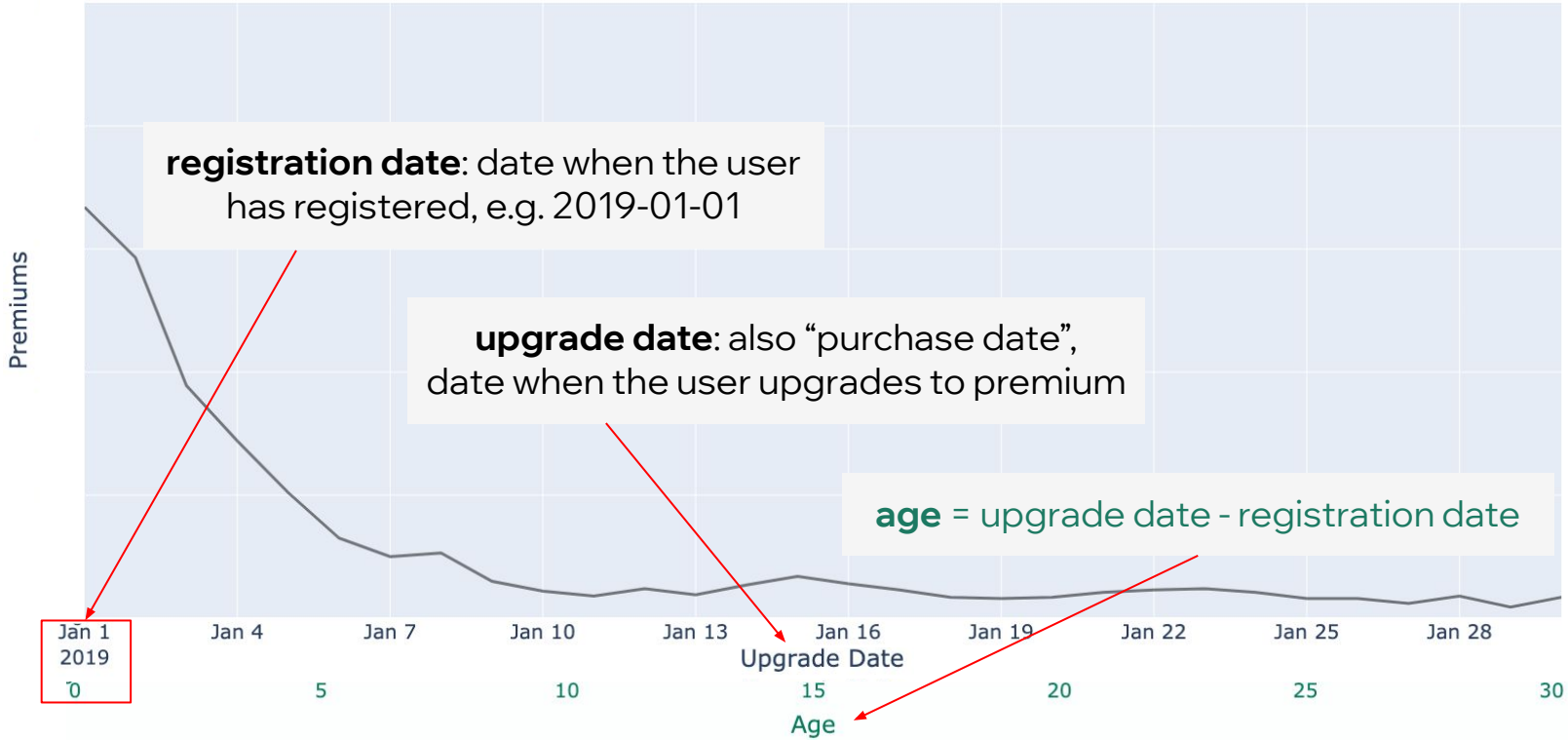
Premiums by upgrade date

for the cohort of registration date: 2019-01-01



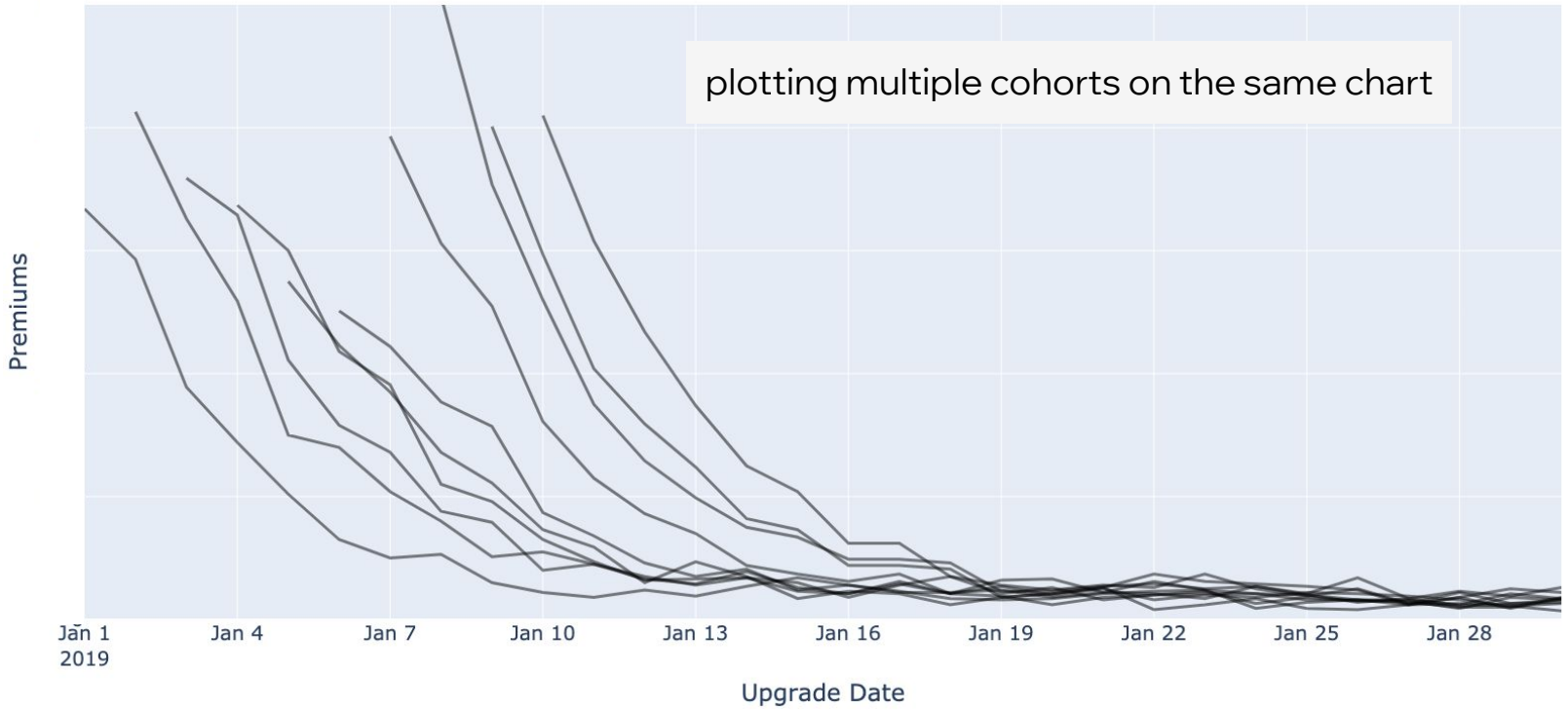
Premiums by upgrade date & age

for the cohort of registration date: 2019-01-01



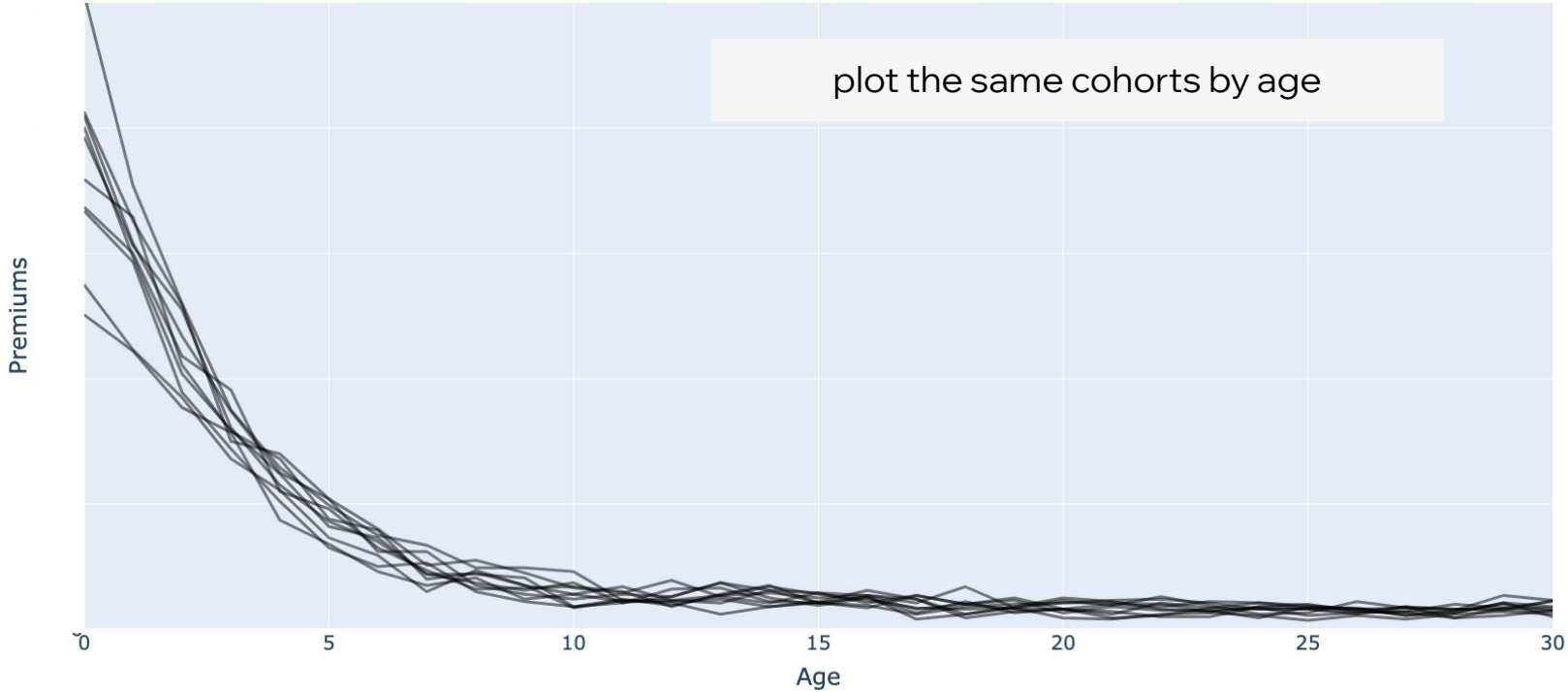
Cohorts behave similarly

they have similar shapes for premiums



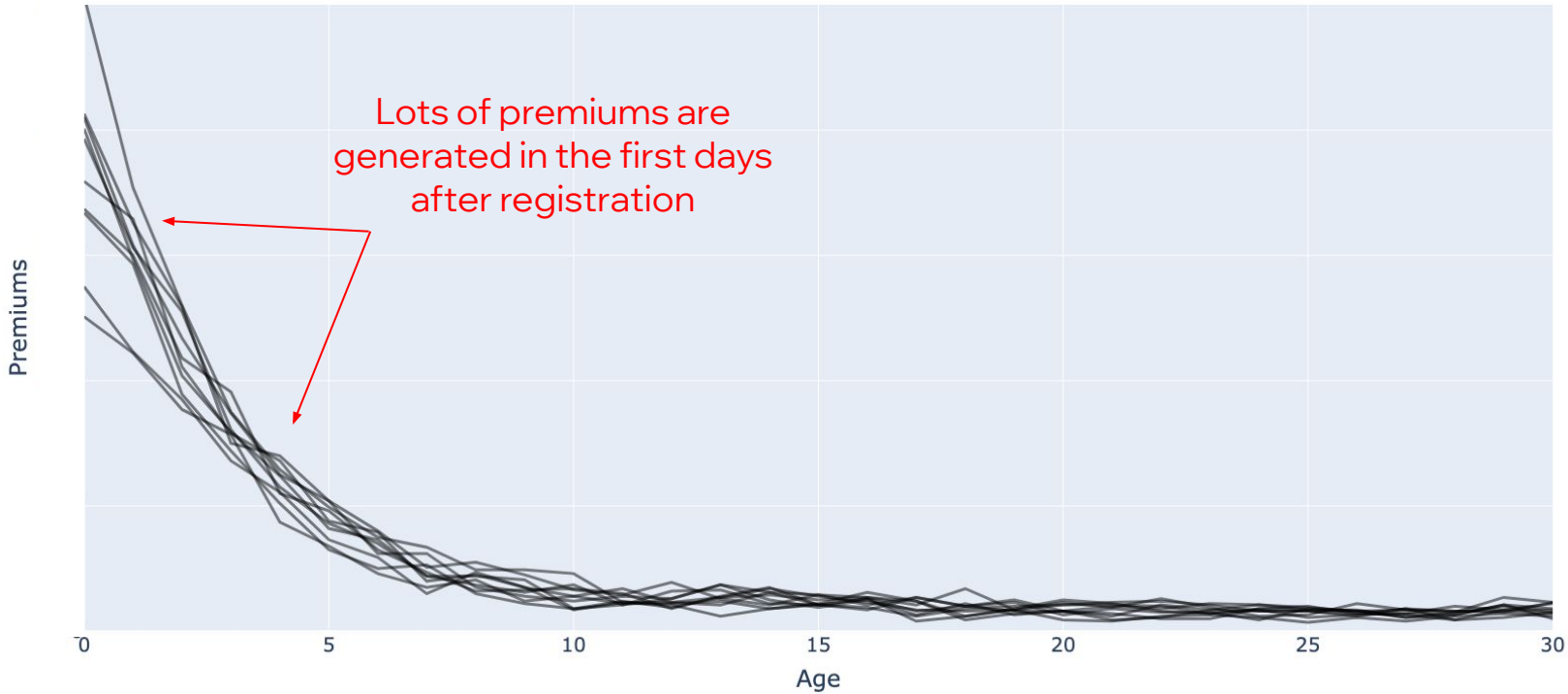
Cohorts behave similarly

they have similar shapes for premiums by age



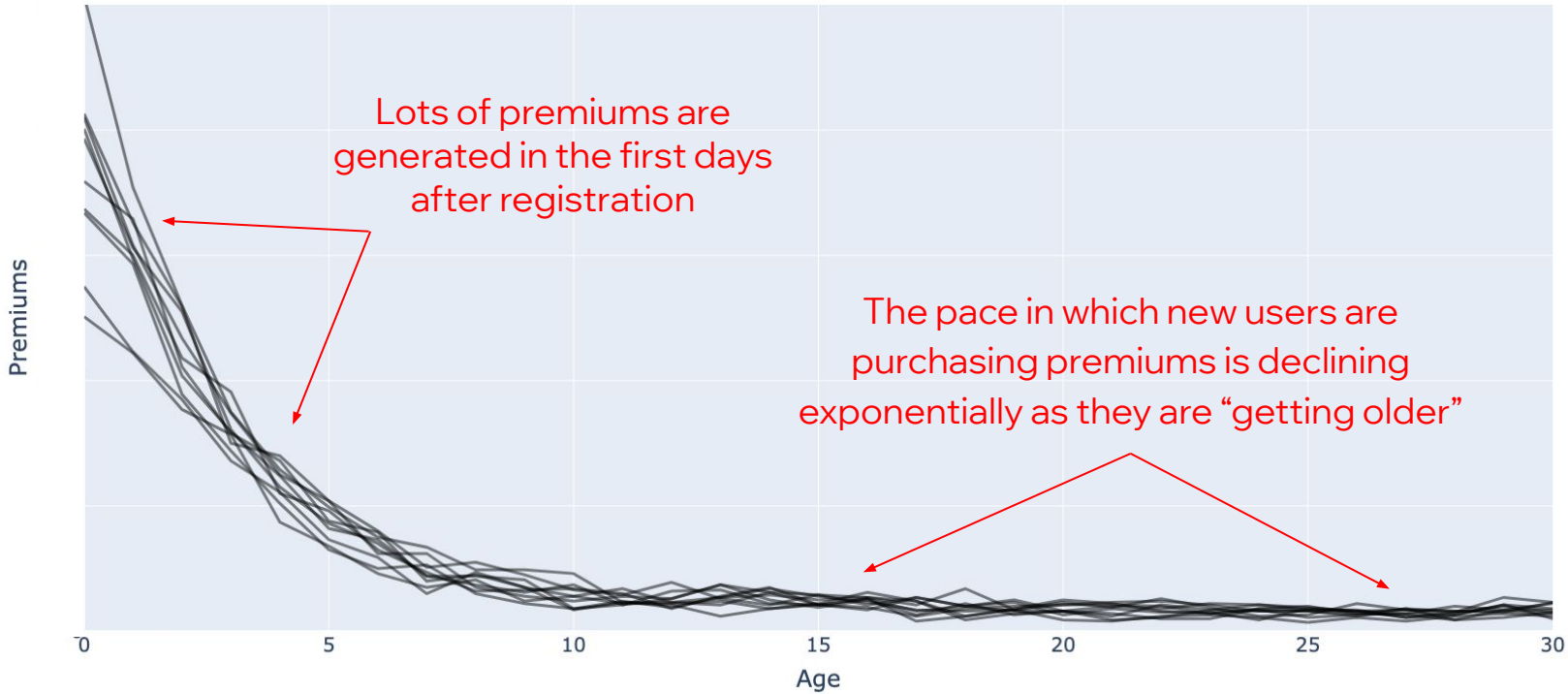
Cohorts behave similarly

they have similar shapes for premiums by age



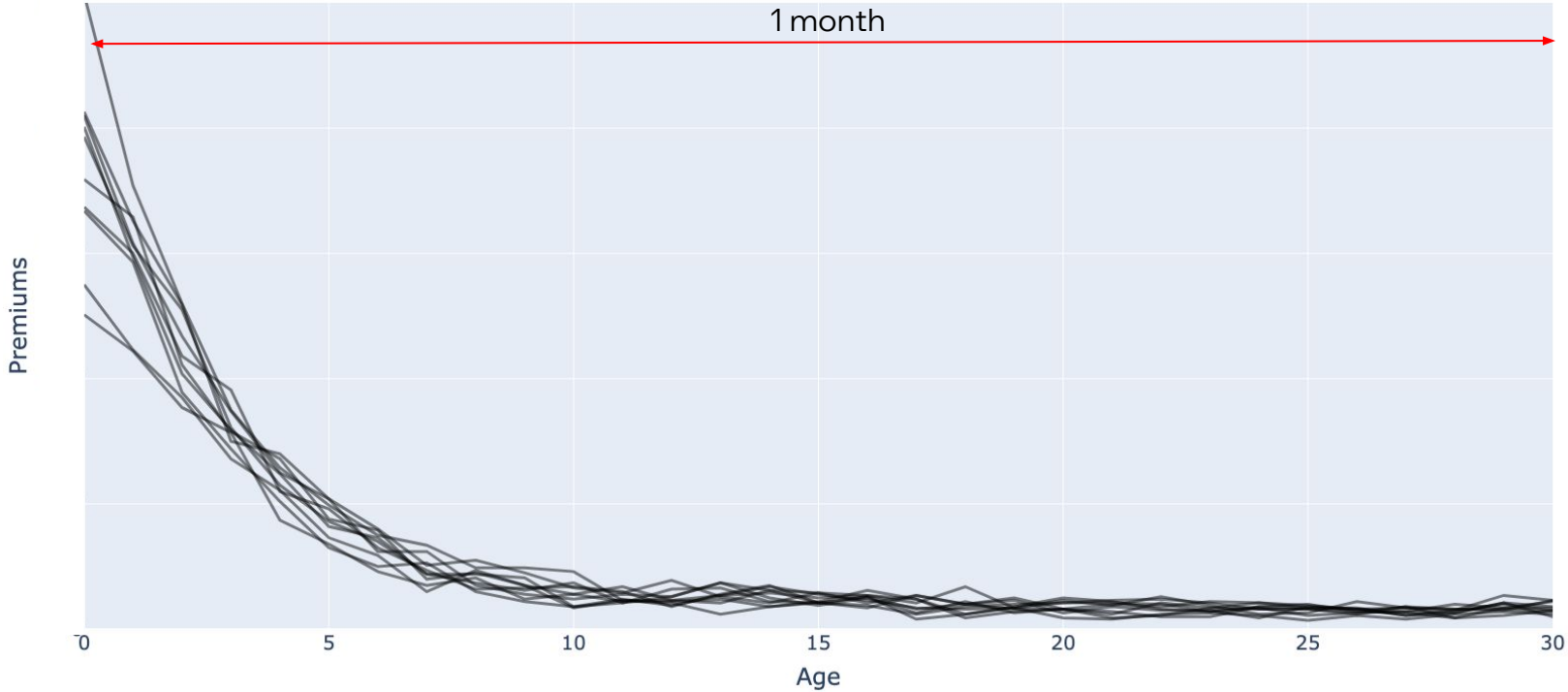
Cohorts behave similarly

they have similar shapes for premiums by age



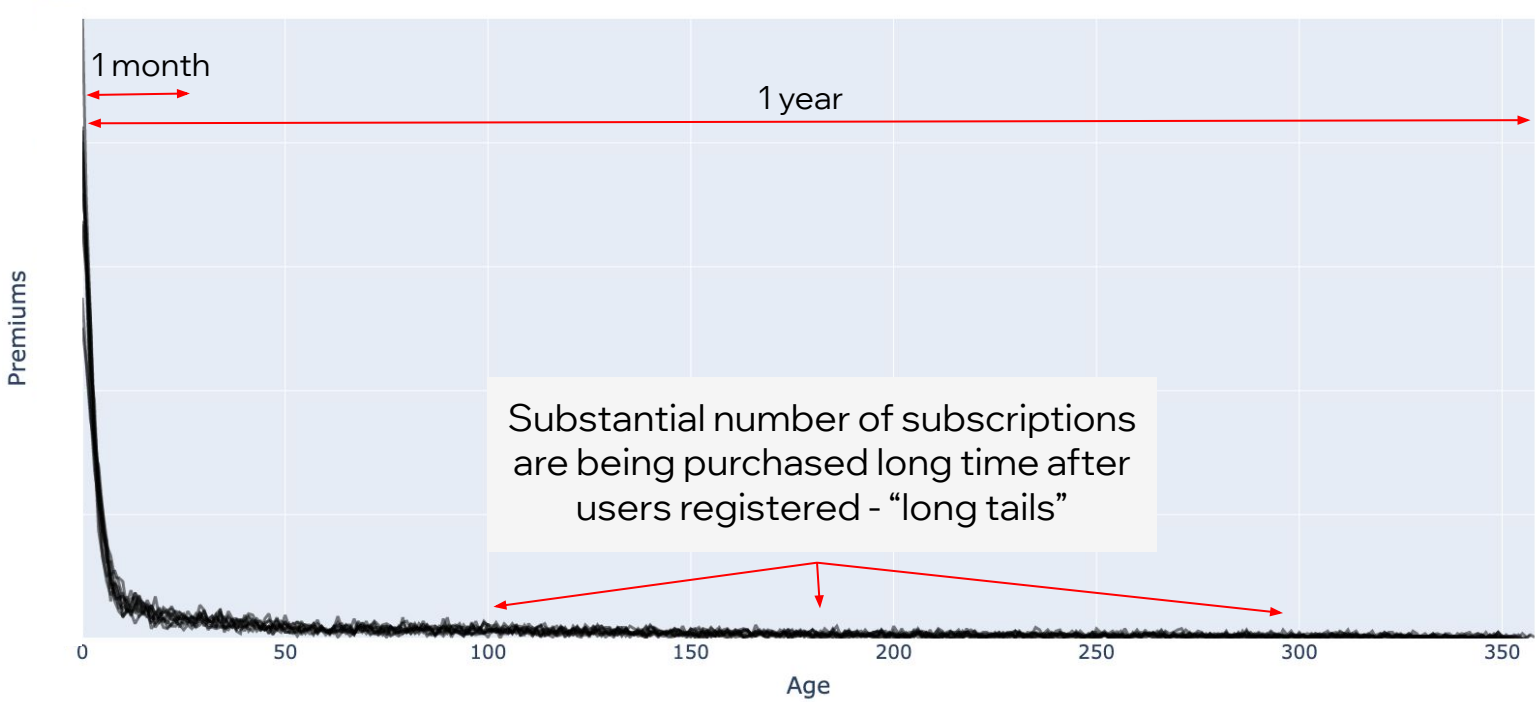
Cohorts behave similarly

they have similar shapes for premiums by age



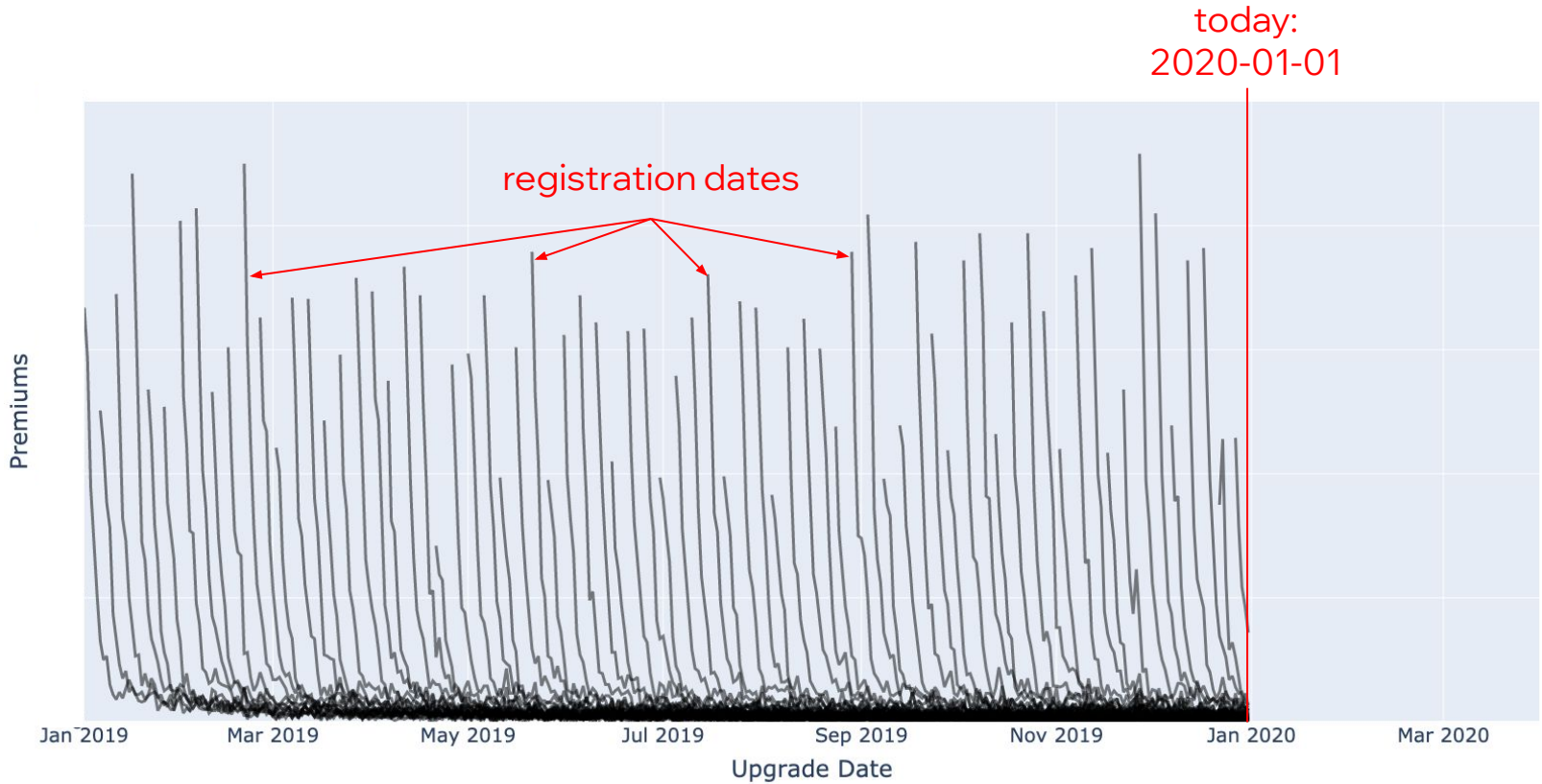
Cohorts have long tails

even for one year ahead they still generate premiums



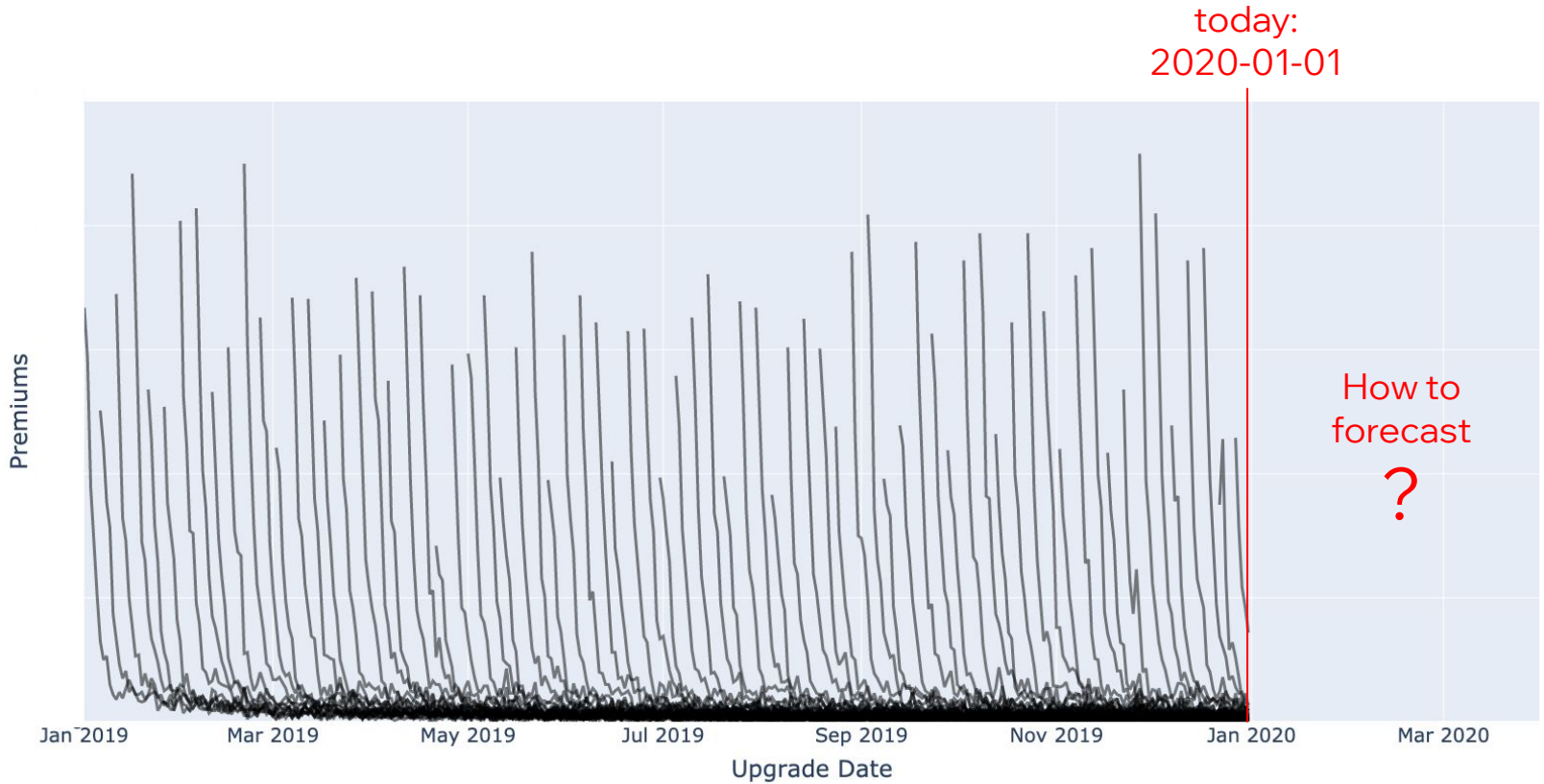
Predicting cohorts

Recent cohorts
registered during last year



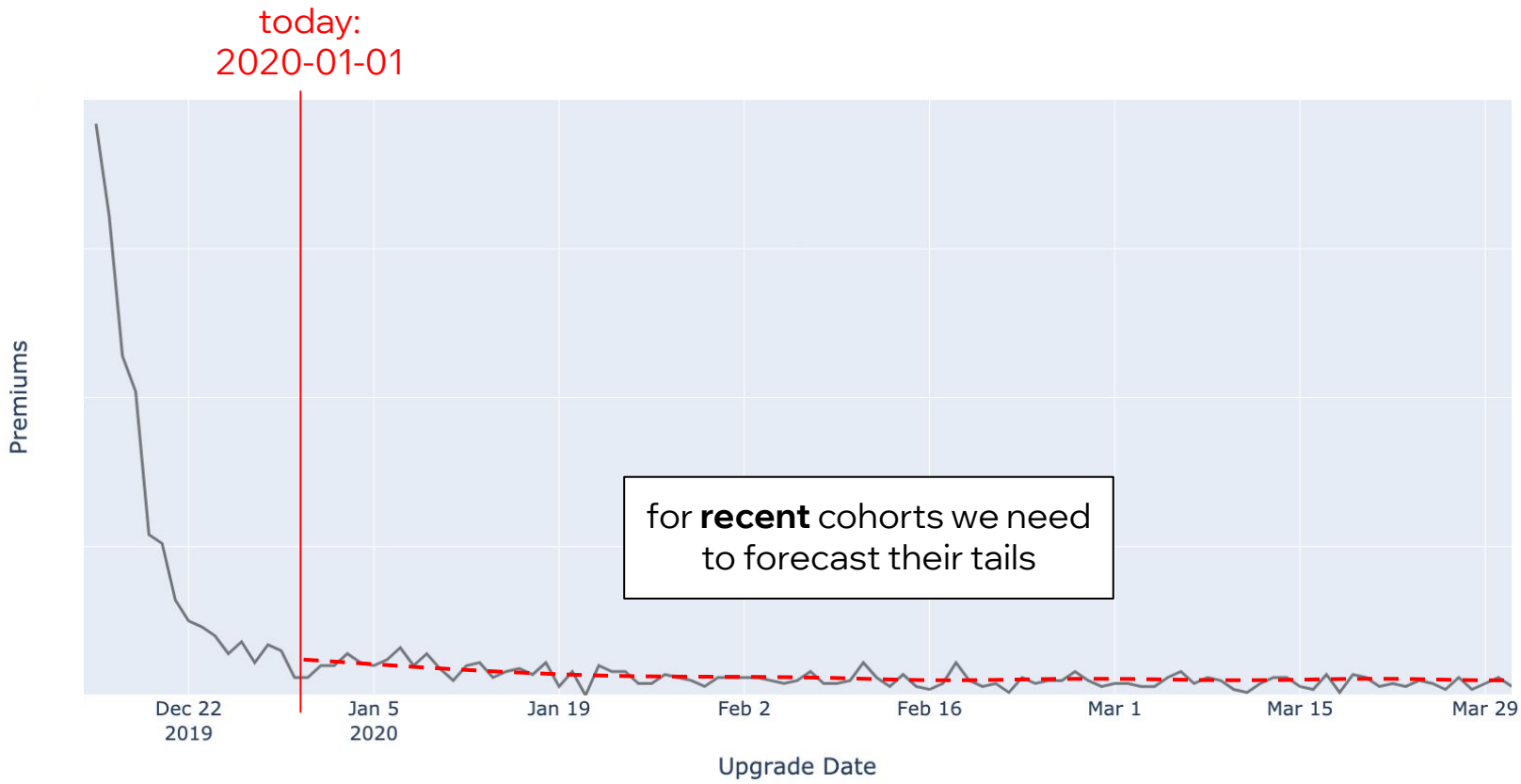
Forecasting one quarter ahead

How to forecast?

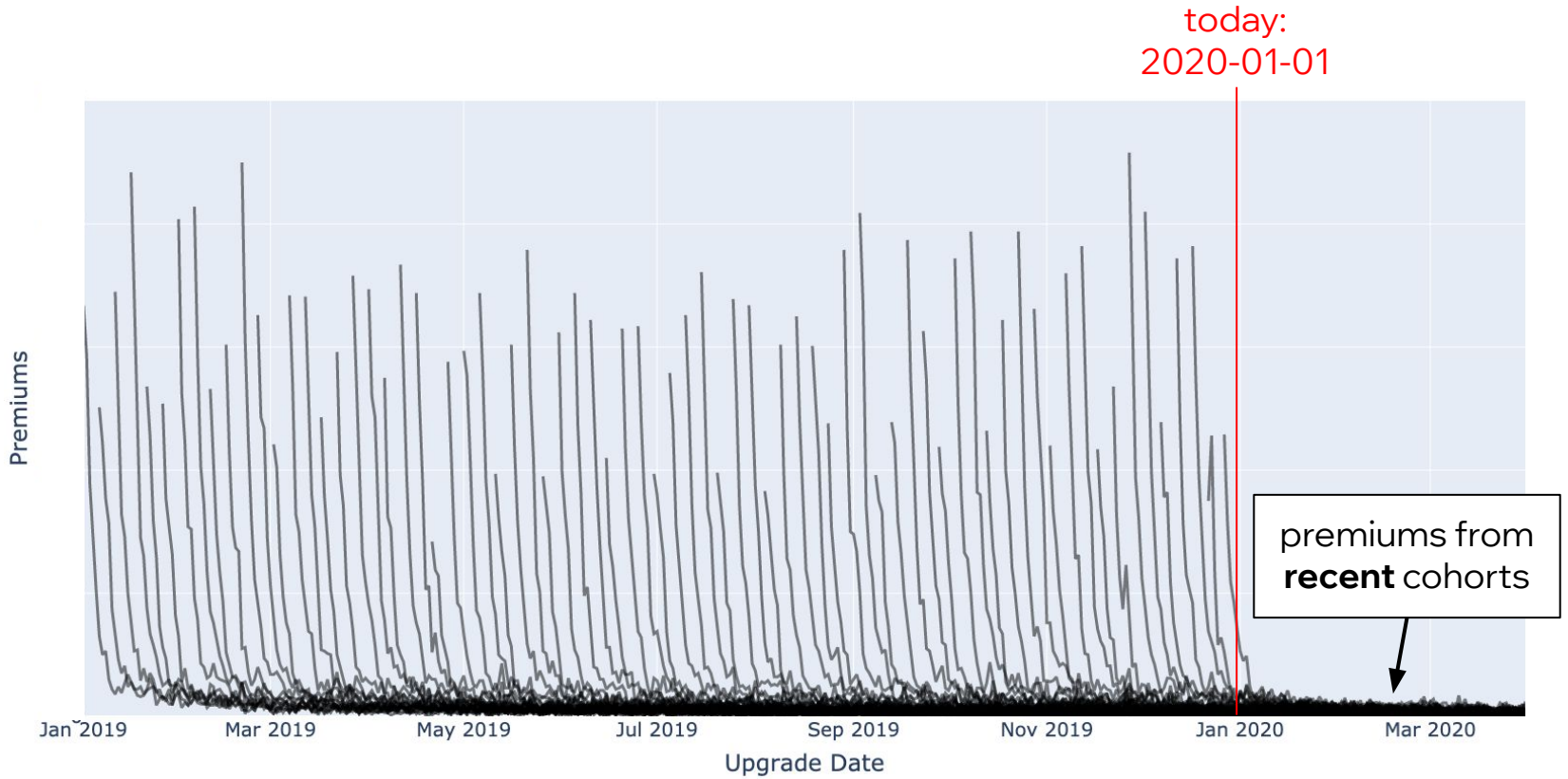


Recent cohorts

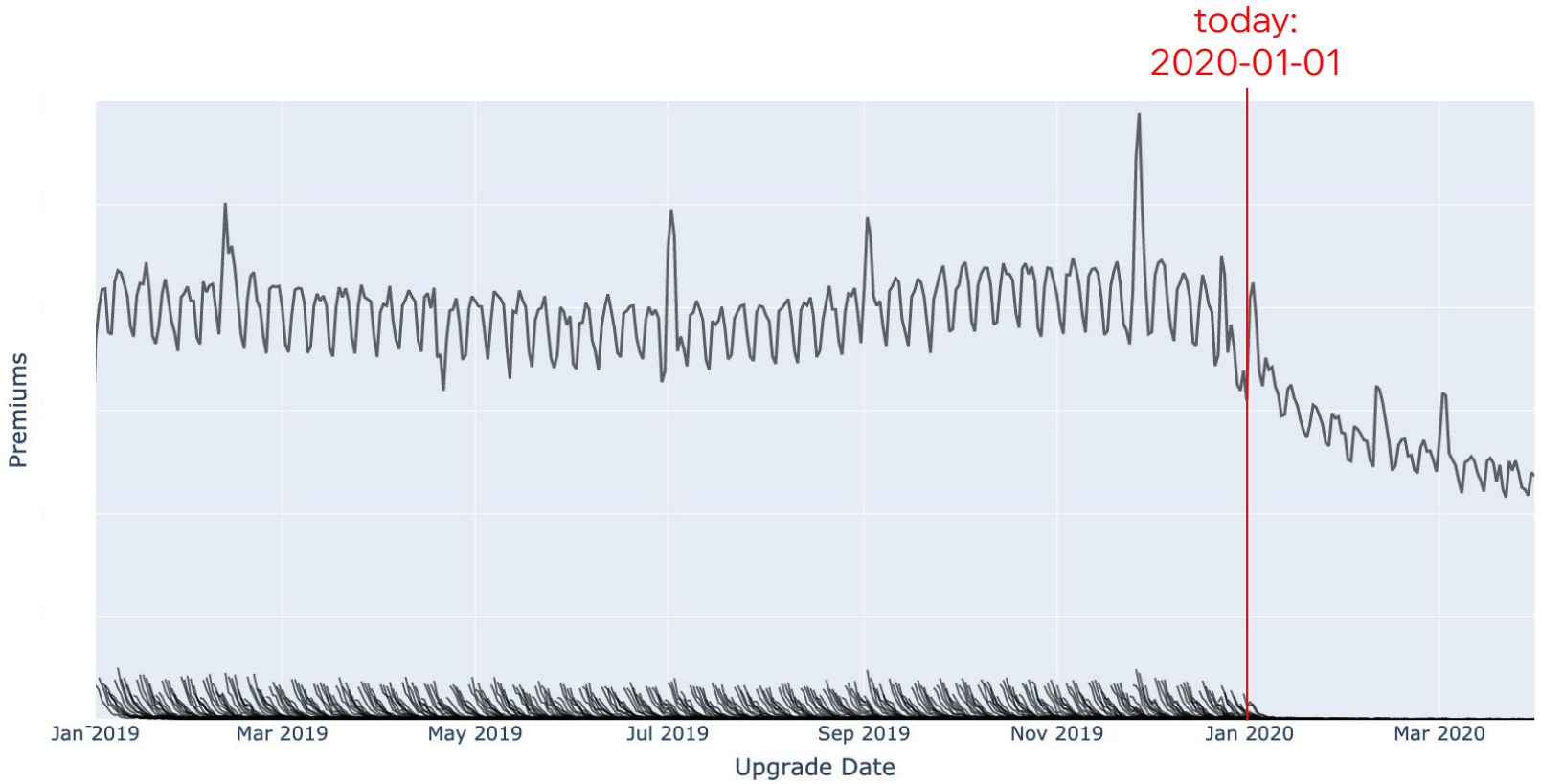
forecast tails one quarter ahead



Recent cohorts
forecast tails one quarter ahead

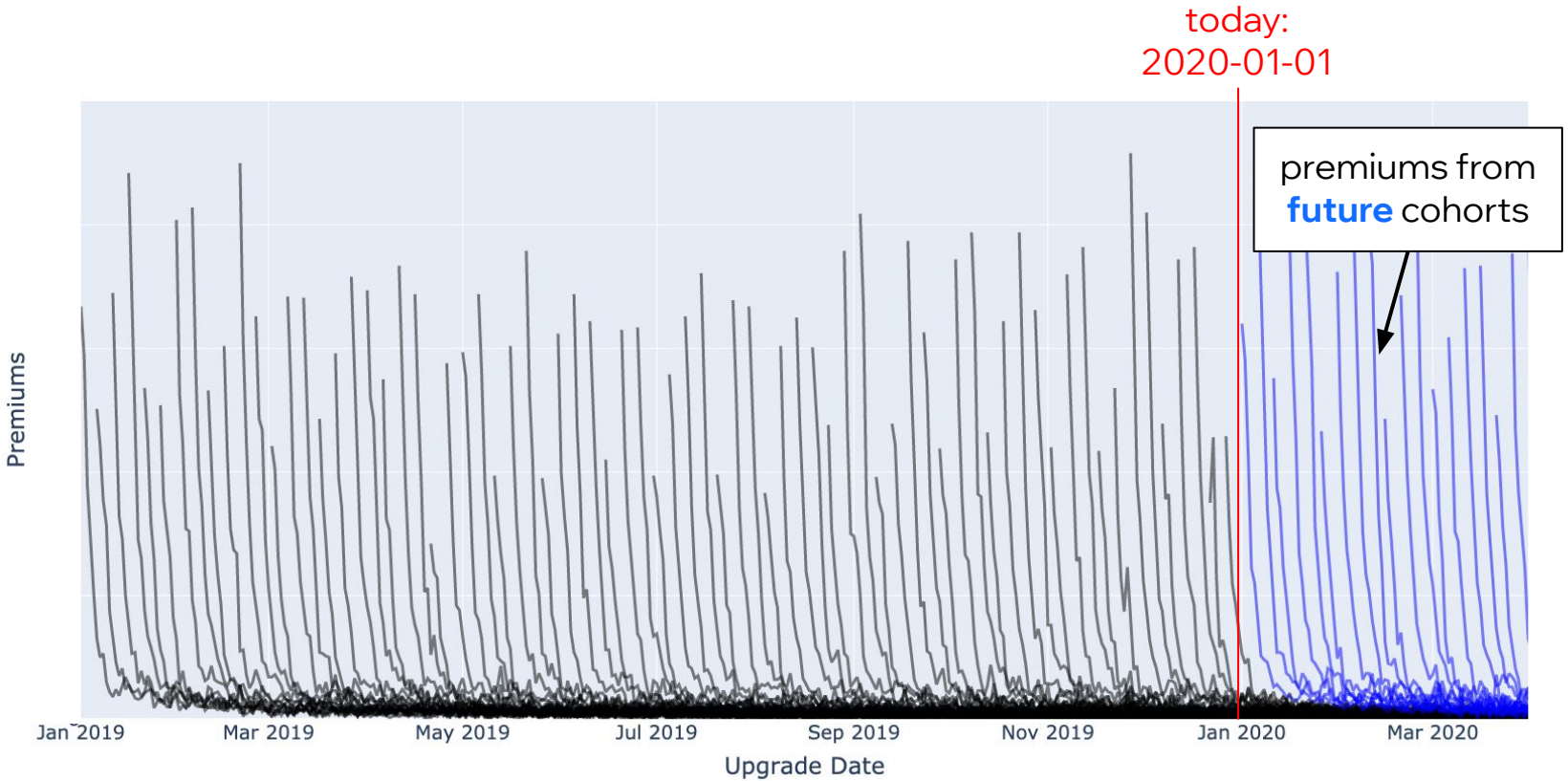


Recent cohorts
aggregated by upgrade date

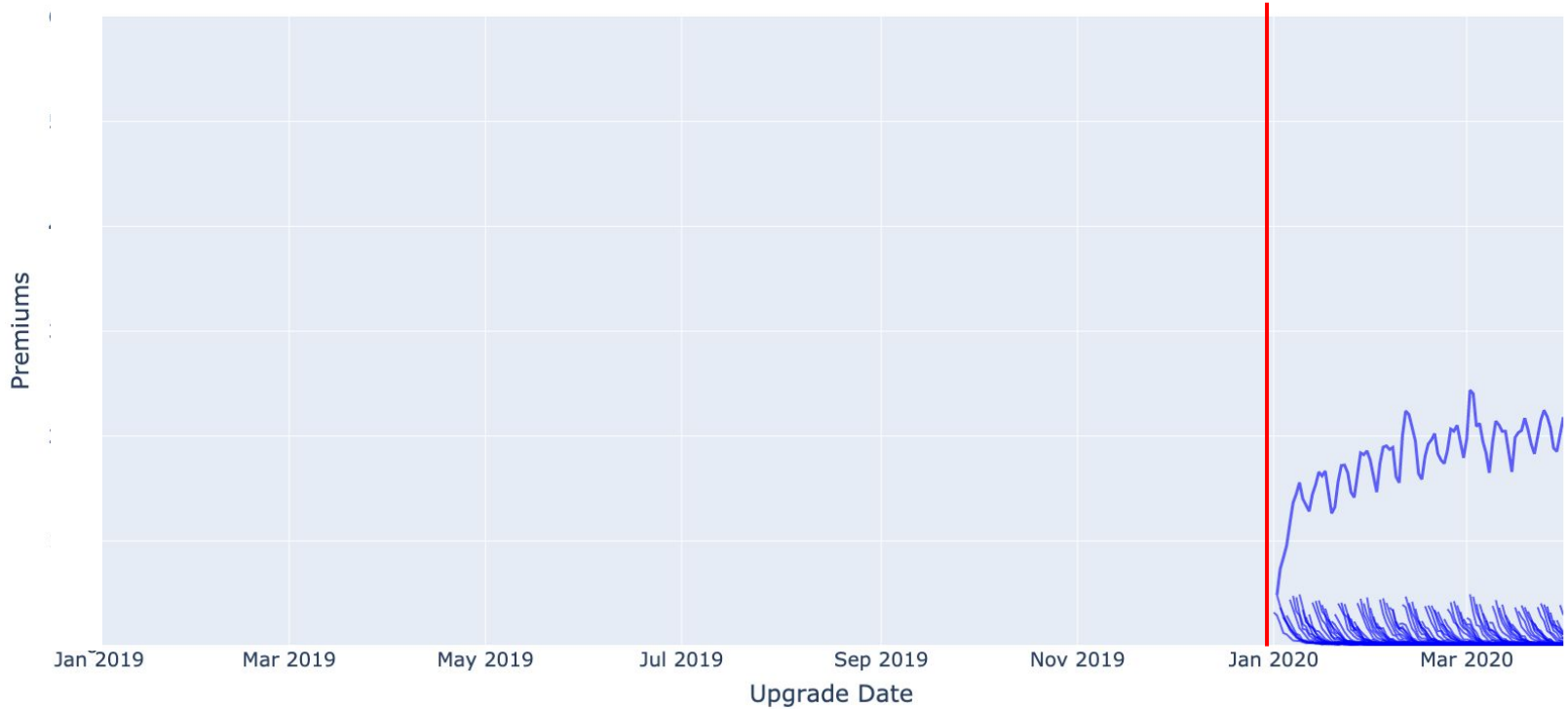


Future cohorts

registered after train date, during the forecast quarter

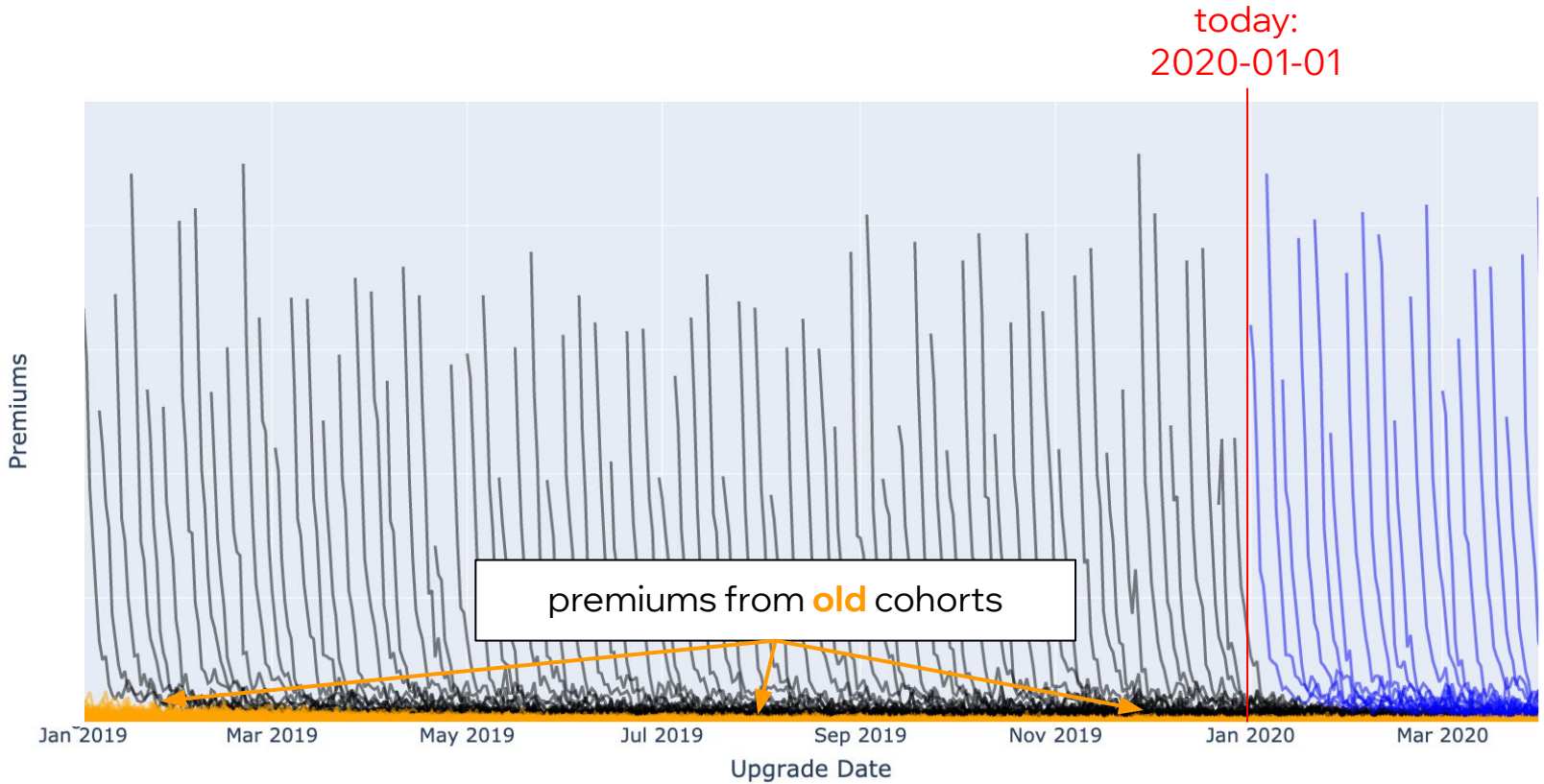


Future cohorts
aggregated by upgrade date



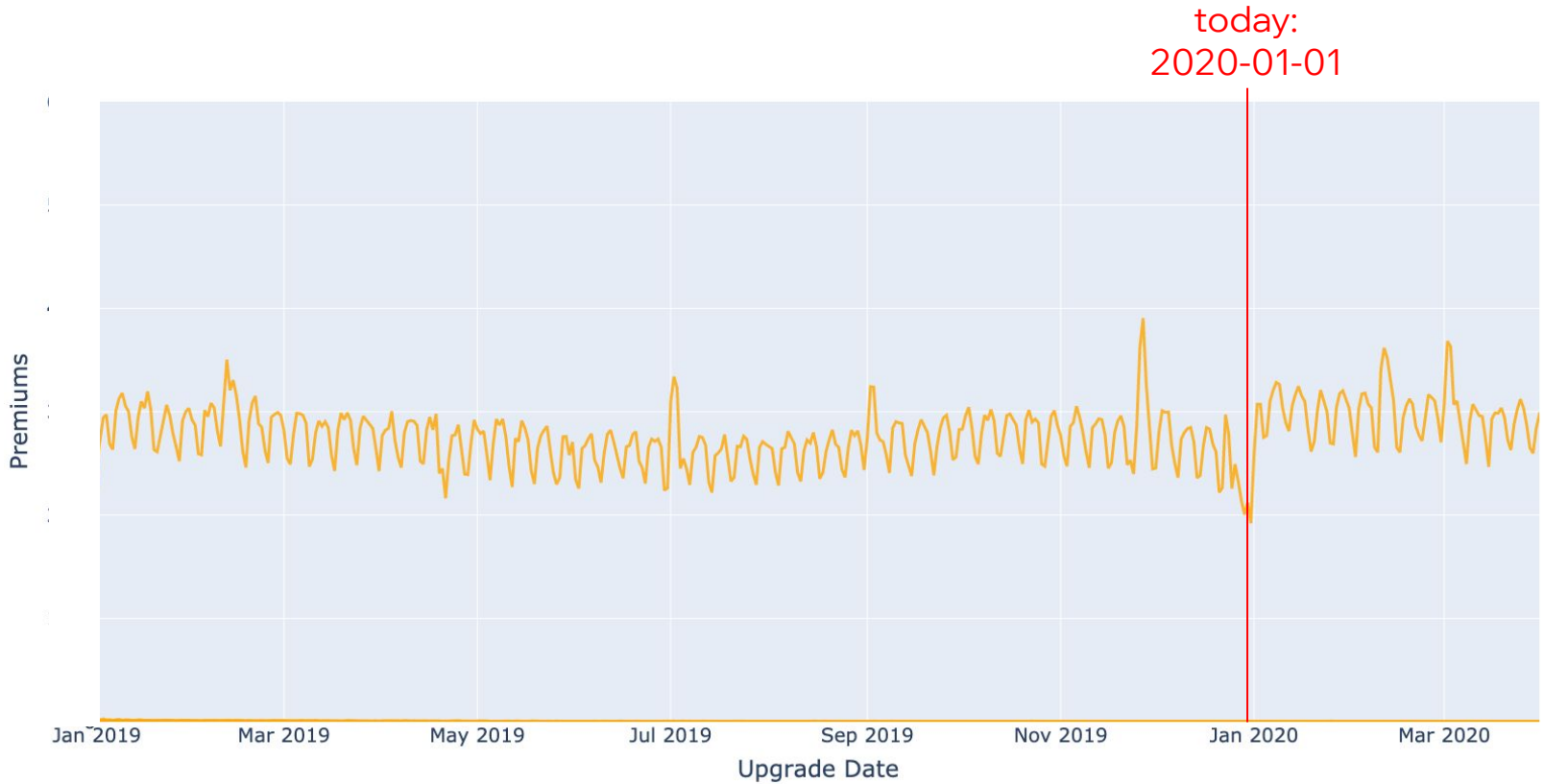
Old cohorts

generate premiums long time after registration

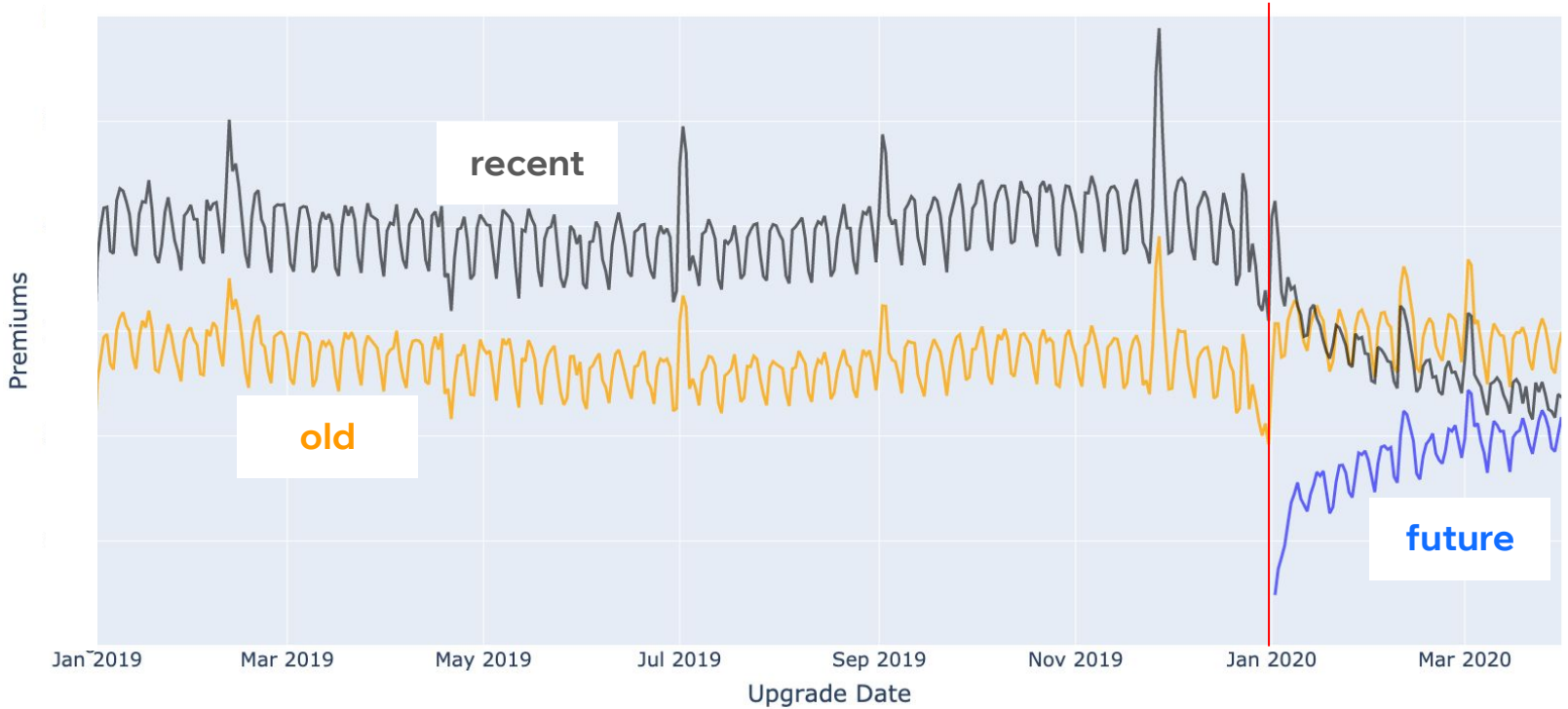


Old cohorts

aggregated by upgrade date

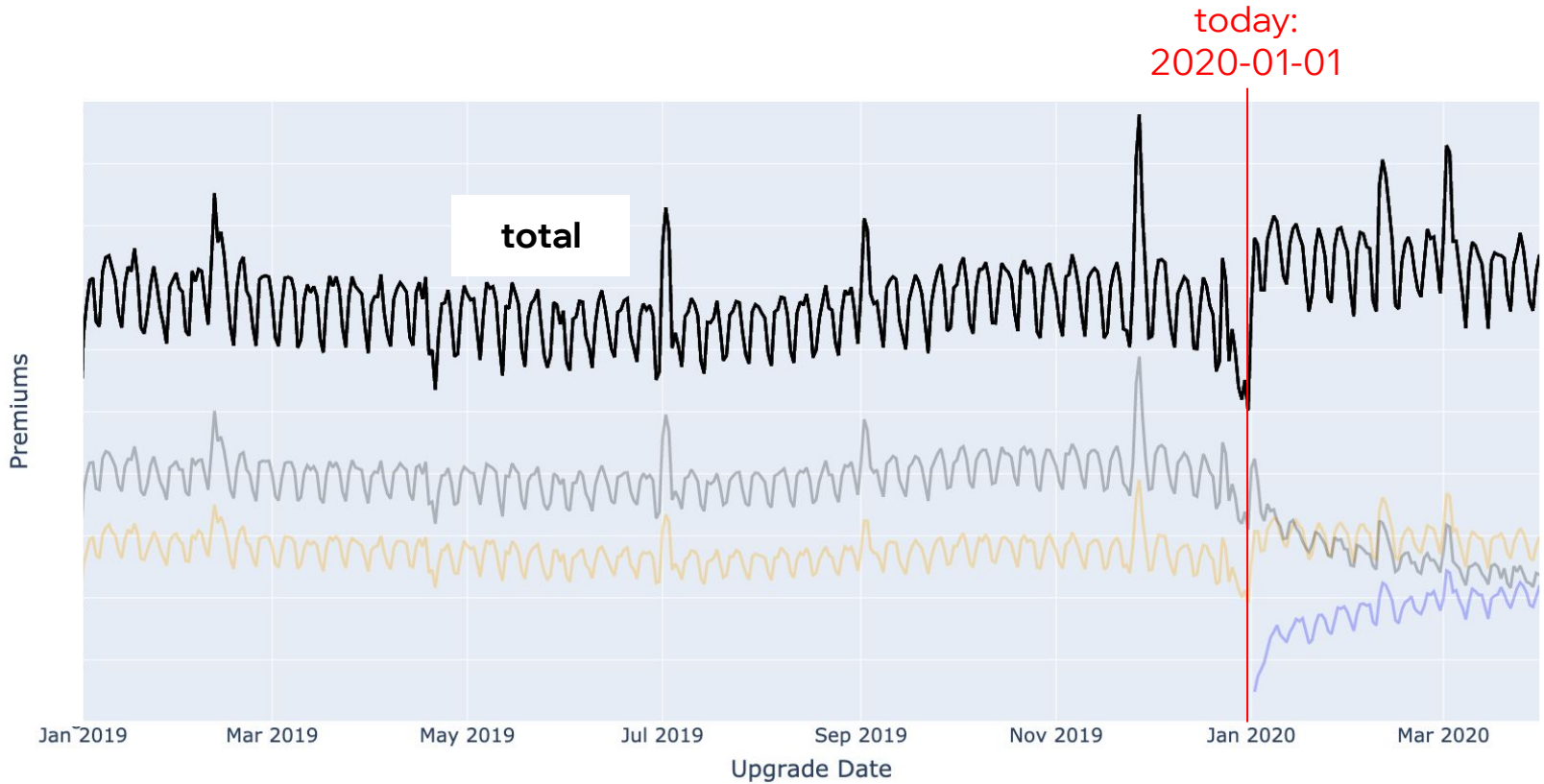


Old, Recent and Future cohorts
comparison

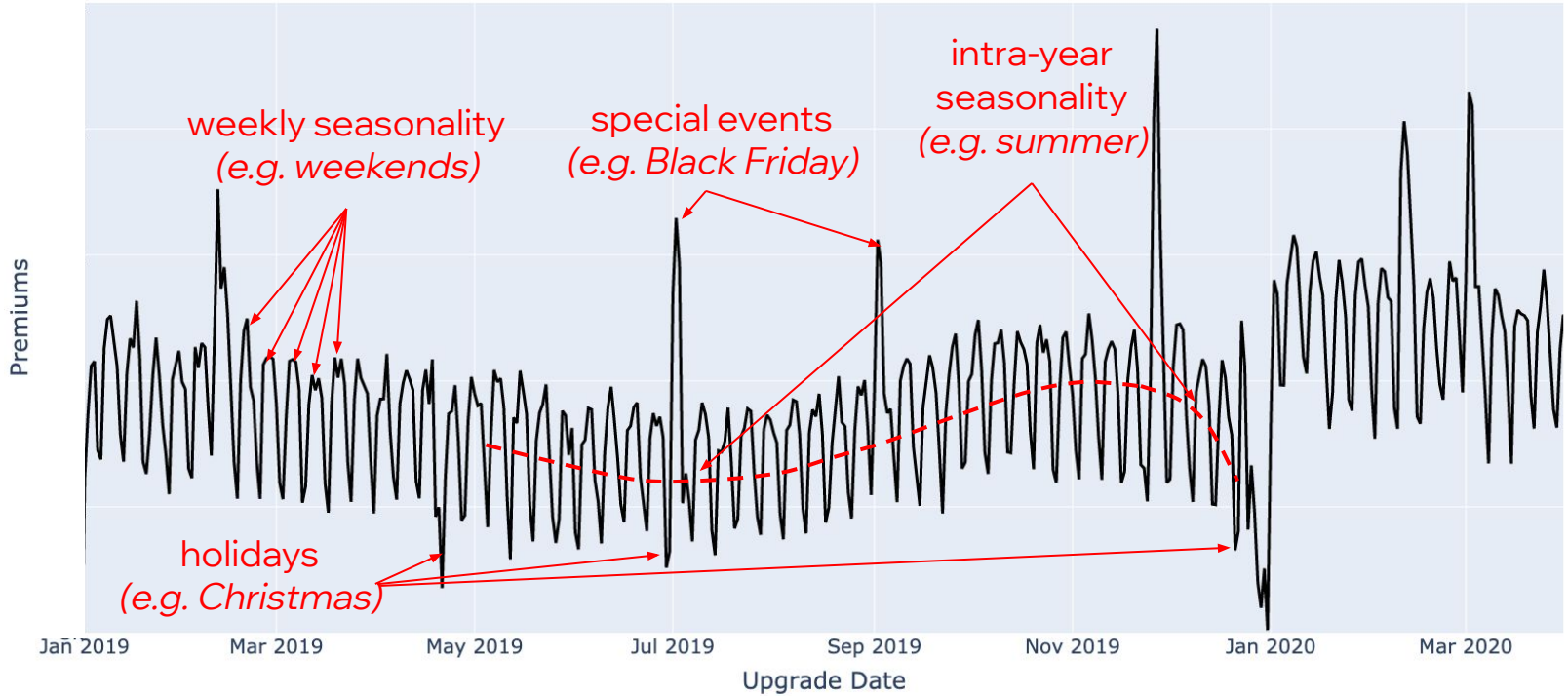


Old, Recent and Future cohorts

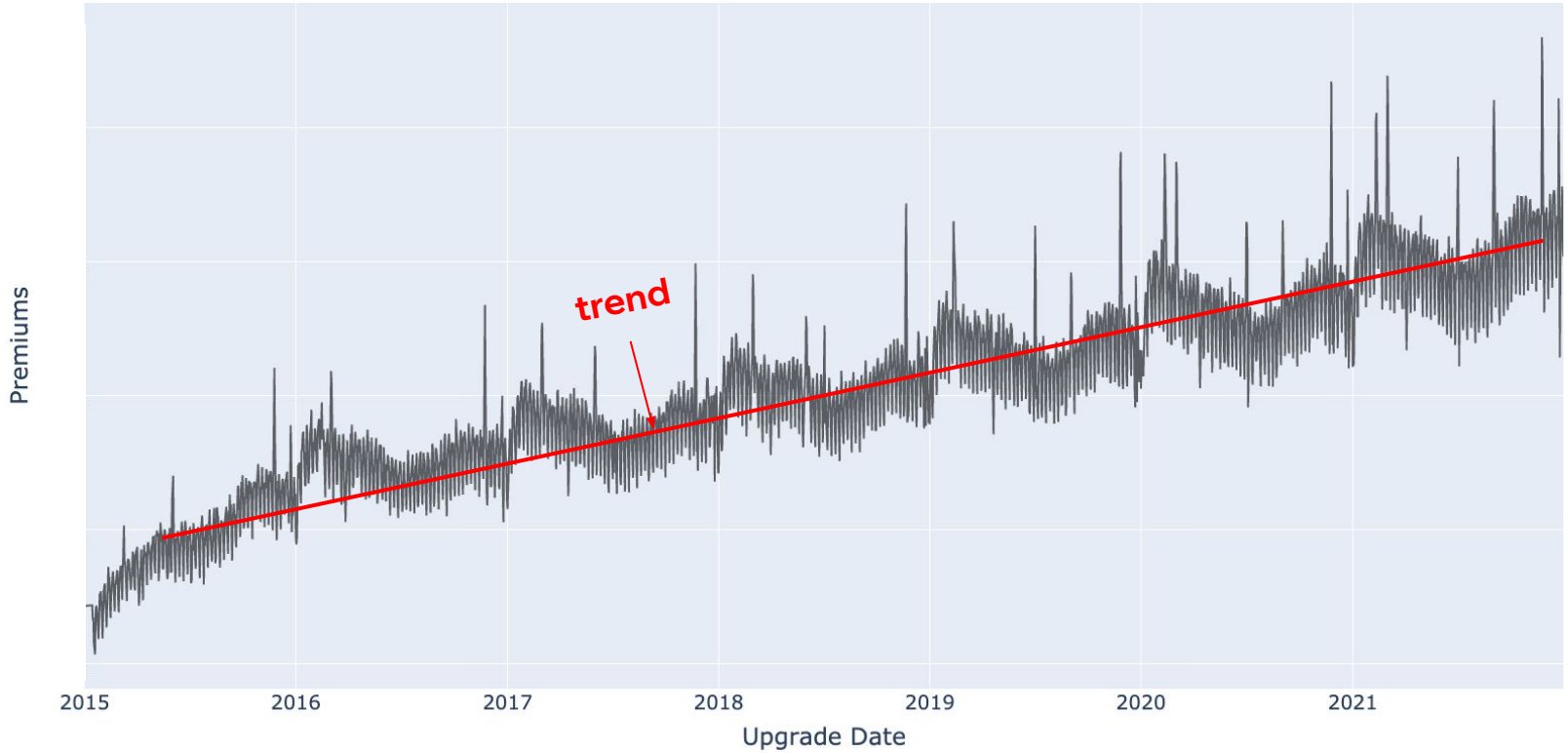
add up to a time-series of total premiums



Time series of premiums
has some interesting properties



Time series of premiums
has some interesting properties



Using the setup for a **cohort-based model** we transform a **time-series** task into a **regression** task.

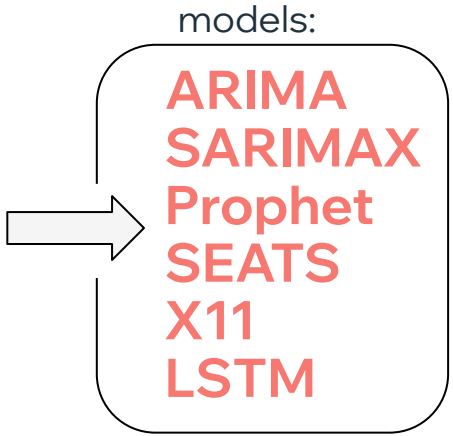
Regression models for cohort-based forecasting

Time-series data and models

date-value pairs + exogenous variables

time-series (date-value pairs) | exogenous variables

	date	premiums	events	holiday
	2019-01-01	234	0	1
	2019-01-02	456	0	0
	2019-01-03	462	0	0
	2019-01-04	1345	1	0
	2019-01-05	1017	1	0
	2019-01-06	962	1	0
	2019-01-07	531	0	0
	2019-01-08	385	0	0
today	...	...	...	...
	2020-01-01	?	0	1
	2020-01-02	?	0	0



*synthetic data

Table data for cohorts

values by double key: registration & upgrade date (age)

double key | target | features

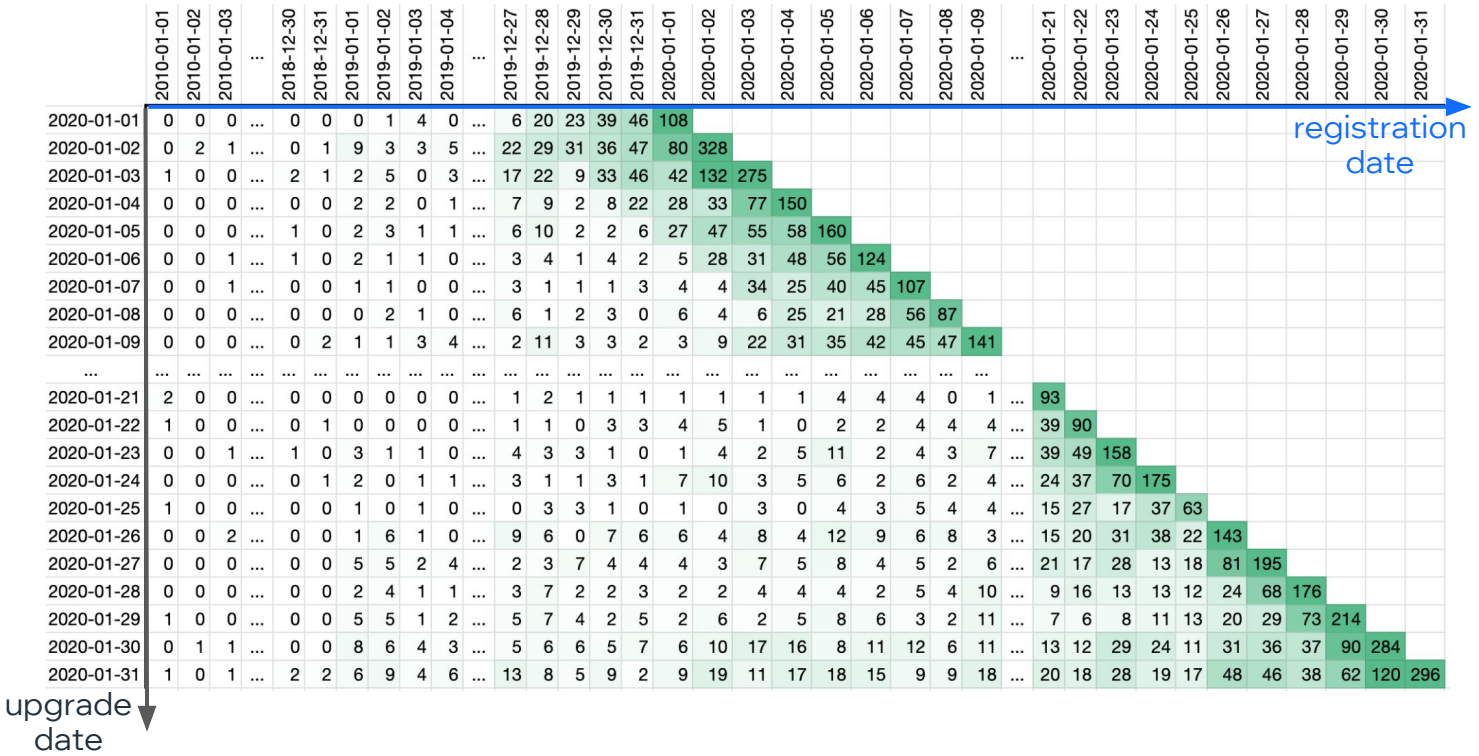
	registration date	upgrade date	premiums	age	events	holiday	s(weekday)
cohort of reg day 2019-01-01	2019-01-01	2019-01-01	200	1	0	1	0.13
	2019-01-01	2019-01-02	138	2	1	0	0.22
	...	...	...	...	...	...	...
	2019-01-01	2020-03-26	2	450	0	0	0.05
cohort of reg day 2019-01-02	2019-01-02	2019-01-02	198	1	1	0	0.22
	2019-01-02	2019-01-03	125	2	1	0	0.34
	...	...	...	...	...	...	...
today	2020-01-01	2020-01-01	?	1	0	0	0.26
	2020-01-01	2020-01-02	?	2	0	0	0.09
	...	...	...	...	...	...	...
	2020-01-01	2021-03-31	?	455	1	0	0.56

Cartesian product of registration dates and upgrade dates (age)

⇒ many rows produced

*synthetic data

Cohorts in triangle by registration and upgrade date

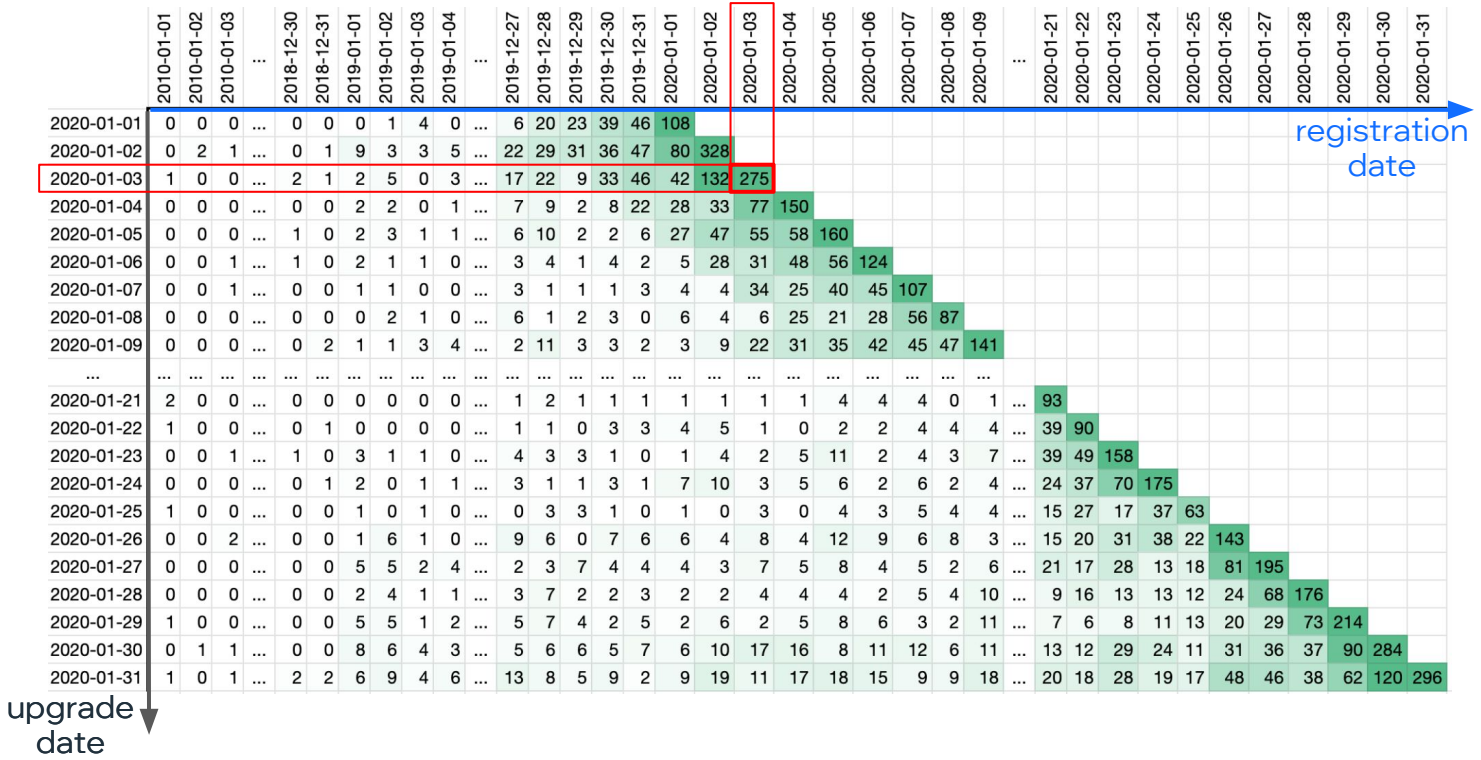


	2010-01-01	2010-01-02	2010-01-03	...	2018-12-30	2018-12-31	2019-01-01	2019-01-02	2019-01-03	2019-01-04	...	2019-12-27	2019-12-28	2019-12-29	2019-12-30	2019-12-31	2020-01-01	2020-01-02	2020-01-03	2020-01-04	2020-01-05	2020-01-06	2020-01-07	2020-01-08	2020-01-09	...	2020-01-21	2020-01-22	2020-01-23	2020-01-24	2020-01-25	2020-01-26	2020-01-27	2020-01-28	2020-01-29	2020-01-30	2020-01-31
2020-01-01	0	0	0	...	0	0	0	1	4	0	...	6	20	23	39	46	108																				
2020-01-02	0	2	1	...	0	1	9	3	3	5	...	22	29	31	36	47	80	328																			
2020-01-03	1	0	0	...	2	1	2	5	0	3	...	17	22	9	33	46	42	132	275																	
2020-01-04	0	0	0	...	0	0	2	2	0	1	...	7	9	2	8	22	28	33	77	150																
2020-01-05	0	0	0	...	1	0	2	3	1	1	...	6	10	2	2	6	27	47	55	58	160															
2020-01-06	0	0	1	...	1	0	2	1	1	0	...	3	4	1	4	2	5	28	31	48	56	124														
2020-01-07	0	0	1	...	0	0	1	1	0	0	...	3	1	1	1	3	4	4	34	25	40	45	107													
2020-01-08	0	0	0	...	0	0	0	2	1	0	...	6	1	2	3	0	6	4	6	25	21	28	56	87												
2020-01-09	0	0	0	...	0	2	1	1	3	4	...	2	11	3	3	2	3	9	22	31	35	42	45	47	141											
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...											
2020-01-21	2	0	0	...	0	0	0	0	0	0	...	1	2	1	1	1	1	1	1	1	4	4	4	0	1	...	93										
2020-01-22	1	0	0	...	0	1	0	0	0	0	...	1	1	0	3	3	4	5	1	0	2	2	4	4	4	...	39	90									
2020-01-23	0	0	1	...	1	0	3	1	1	0	...	4	3	3	1	0	1	4	2	5	11	2	4	3	7	...	39	49	158								
2020-01-24	0	0	0	...	0	1	2	0	1	1	...	3	1	1	3	1	7	10	3	5	6	2	6	2	4	...	24	37	70	175							
2020-01-25	1	0	0	...	0	0	1	0	1	0	...	0	3	3	1	0	1	0	3	0	4	3	5	4	4	...	15	27	17	37	63						
2020-01-26	0	0	2	...	0	0	1	6	1	0	...	9	6	0	7	6	6	4	8	4	12	9	6	8	3	...	15	20	31	38	22	143					
2020-01-27	0	0	0	...	0	0	5	5	2	4	...	2	3	7	4	4	4	3	7	5	8	4	5	2	6	...	21	17	28	13	18	81	195				
2020-01-28	0	0	0	...	0	0	2	4	1	1	...	3	7	2	2	3	2	2	4	4	4	2	5	4	10	...	9	16	13	13	12	24	68	176			
2020-01-29	1	0	0	...	0	0	5	5	1	2	...	5	7	4	2	5	2	6	2	5	8	6	3	2	11	...	7	6	8	11	13	20	29	73	214		
2020-01-30	0	1	1	...	0	0	8	6	4	3	...	5	6	6	5	7	6	10	17	16	8	11	12	6	11	...	13	12	29	24	11	31	36	37	90	284	
2020-01-31	1	0	1	...	2	2	6	9	4	6	...	13	8	5	9	2	9	19	11	17	18	15	9	9	18	...	20	18	28	19	17	48	46	38	62	120	296

*synthetic data

Cohorts in triangle by registration and upgrade date

registration: 2020-01-03
upgrade: 2020-01-03



	2010-01-01	2010-01-02	2010-01-03	...	2018-12-30	2018-12-31	2019-01-01	2019-01-02	2019-01-03	2019-01-04	...	2019-12-27	2019-12-28	2019-12-29	2019-12-30	2019-12-31	2020-01-01	2020-01-02	2020-01-03	2020-01-04	2020-01-05	2020-01-06	2020-01-07	2020-01-08	2020-01-09	...	2020-01-21	2020-01-22	2020-01-23	2020-01-24	2020-01-25	2020-01-26	2020-01-27	2020-01-28	2020-01-29	2020-01-30	2020-01-31
2020-01-01	0	0	0	...	0	0	0	1	4	0	...	6	20	23	39	46	108																				
2020-01-02	0	2	1	...	0	1	9	3	3	5	...	22	29	31	36	47	80	328																			
2020-01-03	1	0	0	...	2	1	2	5	0	3	...	17	22	9	33	46	42	132	275																		
2020-01-04	0	0	0	...	0	0	2	2	0	1	...	7	9	2	8	22	28	33	77	150																	
2020-01-05	0	0	0	...	1	0	2	3	1	1	...	6	10	2	2	6	27	47	55	58	160																
2020-01-06	0	0	1	...	1	0	2	1	1	0	...	3	4	1	4	2	5	28	31	48	56	124															
2020-01-07	0	0	1	...	0	0	1	1	0	0	...	3	1	1	1	3	4	4	34	25	40	45	107														
2020-01-08	0	0	0	...	0	0	0	2	1	0	...	6	1	2	3	0	6	4	6	25	21	28	56	87													
2020-01-09	0	0	0	...	0	2	1	1	3	4	...	2	11	3	3	2	3	9	22	31	35	42	45	47	141												
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...											
2020-01-21	2	0	0	...	0	0	0	0	0	0	...	1	2	1	1	1	1	1	1	1	4	4	4	0	1	...	93										
2020-01-22	1	0	0	...	0	1	0	0	0	0	...	1	1	0	3	3	4	5	1	0	2	2	4	4	4	...	39	90									
2020-01-23	0	0	1	...	1	0	3	1	1	0	...	4	3	3	1	0	1	4	2	5	11	2	4	3	7	...	39	49	158								
2020-01-24	0	0	0	...	0	1	2	0	1	1	...	3	1	1	3	1	7	10	3	5	6	2	6	2	4	...	24	37	70	175							
2020-01-25	1	0	0	...	0	0	1	0	1	0	...	0	3	3	1	0	1	0	3	0	4	3	5	4	4	...	15	27	17	37	63						
2020-01-26	0	0	2	...	0	0	1	6	1	0	...	9	6	0	7	6	6	4	8	4	12	9	6	8	3	...	15	20	31	38	22	143					
2020-01-27	0	0	0	...	0	0	5	5	2	4	...	2	3	7	4	4	4	3	7	5	8	4	5	2	6	...	21	17	28	13	18	81	195				
2020-01-28	0	0	0	...	0	0	2	4	1	1	...	3	7	2	2	3	2	2	4	4	4	2	5	4	10	...	9	16	13	13	12	24	68	176			
2020-01-29	1	0	0	...	0	0	5	5	1	2	...	5	7	4	2	5	2	6	2	5	8	6	3	2	11	...	7	6	8	11	13	20	29	73	214		
2020-01-30	0	1	1	...	0	0	8	6	4	3	...	5	6	6	5	7	6	10	17	16	8	11	12	6	11	...	13	12	29	24	11	31	36	37	90	284	
2020-01-31	1	0	1	...	2	2	6	9	4	6	...	13	8	5	9	2	9	19	11	17	18	15	9	9	18	...	20	18	28	19	17	48	46	38	62	120	296

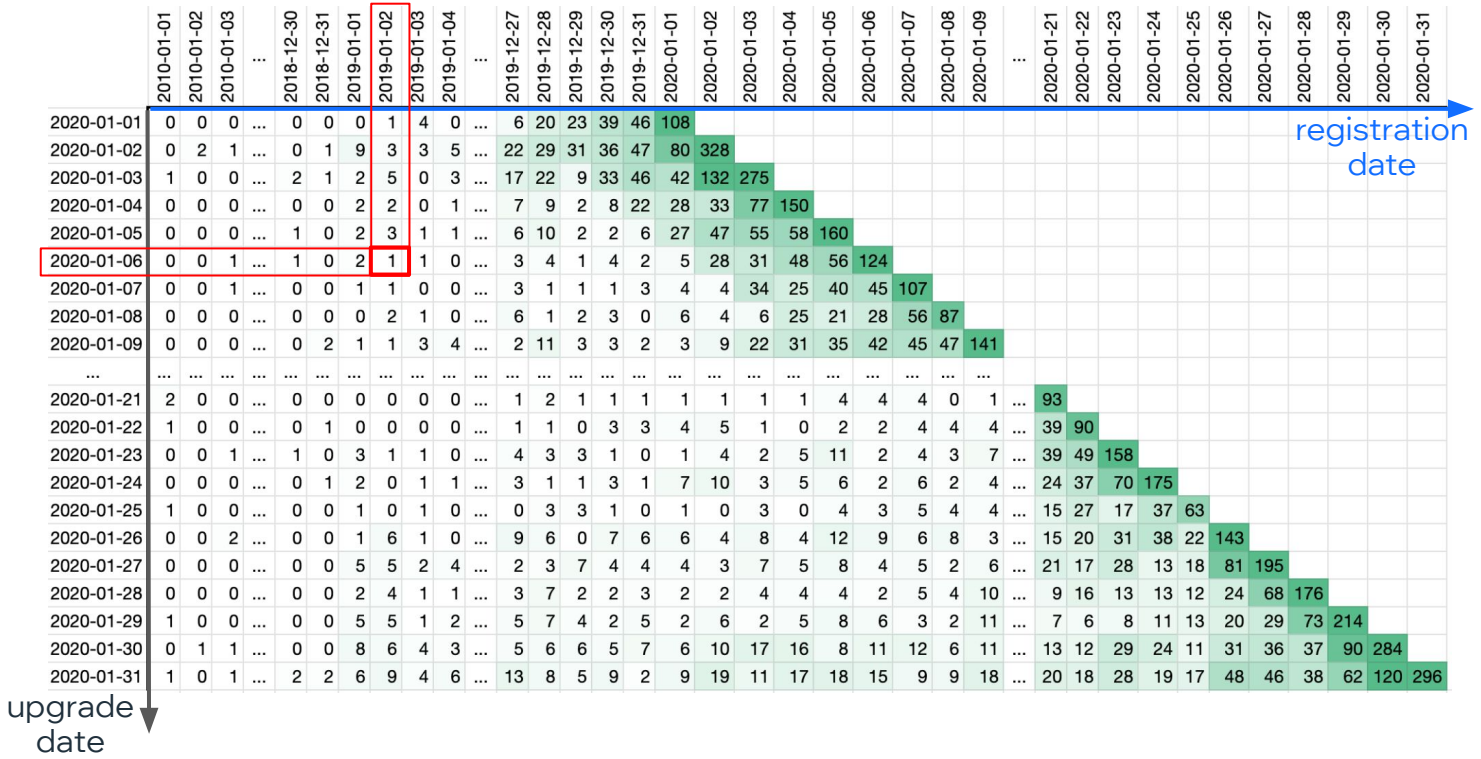
registration date

upgrade date

*synthetic data

Cohorts in triangle by registration and upgrade date

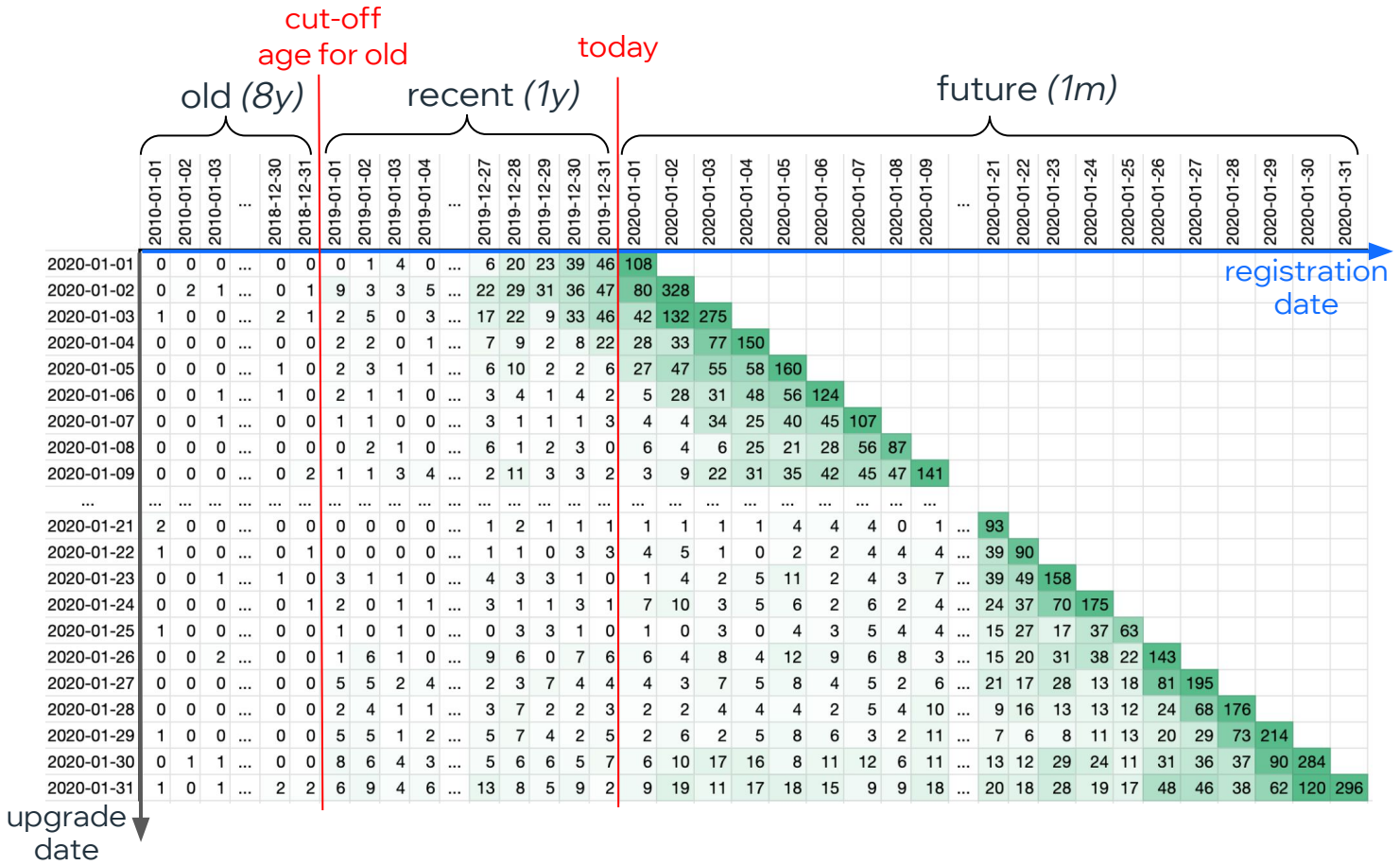
registration: 2019-01-02
upgrade: 2020-01-06



	2010-01-01	2010-01-02	2010-01-03	...	2018-12-30	2018-12-31	2019-01-01	2019-01-02	2019-01-03	2019-01-04	...	2019-12-27	2019-12-28	2019-12-29	2019-12-30	2019-12-31	2020-01-01	2020-01-02	2020-01-03	2020-01-04	2020-01-05	2020-01-06	2020-01-07	2020-01-08	2020-01-09	...	2020-01-21	2020-01-22	2020-01-23	2020-01-24	2020-01-25	2020-01-26	2020-01-27	2020-01-28	2020-01-29	2020-01-30	2020-01-31
2020-01-01	0	0	0	...	0	0	0	1	4	0	...	6	20	23	39	46	108																				
2020-01-02	0	2	1	...	0	1	9	3	3	5	...	22	29	31	36	47	80	328																			
2020-01-03	1	0	0	...	2	1	2	5	0	3	...	17	22	9	33	46	42	132	275																		
2020-01-04	0	0	0	...	0	0	2	2	0	1	...	7	9	2	8	22	28	33	77	150																	
2020-01-05	0	0	0	...	1	0	2	3	1	1	...	6	10	2	2	6	27	47	55	58	160																
2020-01-06	0	0	1	...	1	0	2	1	1	0	...	3	4	1	4	2	5	28	31	48	56	124															
2020-01-07	0	0	1	...	0	0	1	1	0	0	...	3	1	1	1	3	4	4	34	25	40	45	107														
2020-01-08	0	0	0	...	0	0	0	2	1	0	...	6	1	2	3	0	6	4	6	25	21	28	56	87													
2020-01-09	0	0	0	...	0	2	1	1	3	4	...	2	11	3	3	2	3	9	22	31	35	42	45	47	141												
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...											
2020-01-21	2	0	0	...	0	0	0	0	0	0	...	1	2	1	1	1	1	1	1	1	4	4	4	0	1	...	93										
2020-01-22	1	0	0	...	0	1	0	0	0	0	...	1	1	0	3	3	4	5	1	0	2	2	4	4	4	...	39	90									
2020-01-23	0	0	1	...	1	0	3	1	1	0	...	4	3	3	1	0	1	4	2	5	11	2	4	3	7	...	39	49	158								
2020-01-24	0	0	0	...	0	1	2	0	1	1	...	3	1	1	3	1	7	10	3	5	6	2	6	2	4	...	24	37	70	175							
2020-01-25	1	0	0	...	0	0	1	0	1	0	...	0	3	3	1	0	1	0	3	0	4	3	5	4	4	...	15	27	17	37	63						
2020-01-26	0	0	2	...	0	0	1	6	1	0	...	9	6	0	7	6	6	4	8	4	12	9	6	8	3	...	15	20	31	38	22	143					
2020-01-27	0	0	0	...	0	0	5	5	2	4	...	2	3	7	4	4	4	3	7	5	8	4	5	2	6	...	21	17	28	13	18	81	195				
2020-01-28	0	0	0	...	0	0	2	4	1	1	...	3	7	2	2	3	2	2	4	4	4	2	5	4	10	...	9	16	13	13	12	24	68	176			
2020-01-29	1	0	0	...	0	0	5	5	1	2	...	5	7	4	2	5	2	6	2	5	8	6	3	2	11	...	7	6	8	11	13	20	29	73	214		
2020-01-30	0	1	1	...	0	0	8	6	4	3	...	5	6	6	5	7	6	10	17	16	8	11	12	6	11	...	13	12	29	24	11	31	36	37	90	284	
2020-01-31	1	0	1	...	2	2	6	9	4	6	...	13	8	5	9	2	9	19	11	17	18	15	9	9	18	...	20	18	28	19	17	48	46	38	62	120	296

*synthetic data

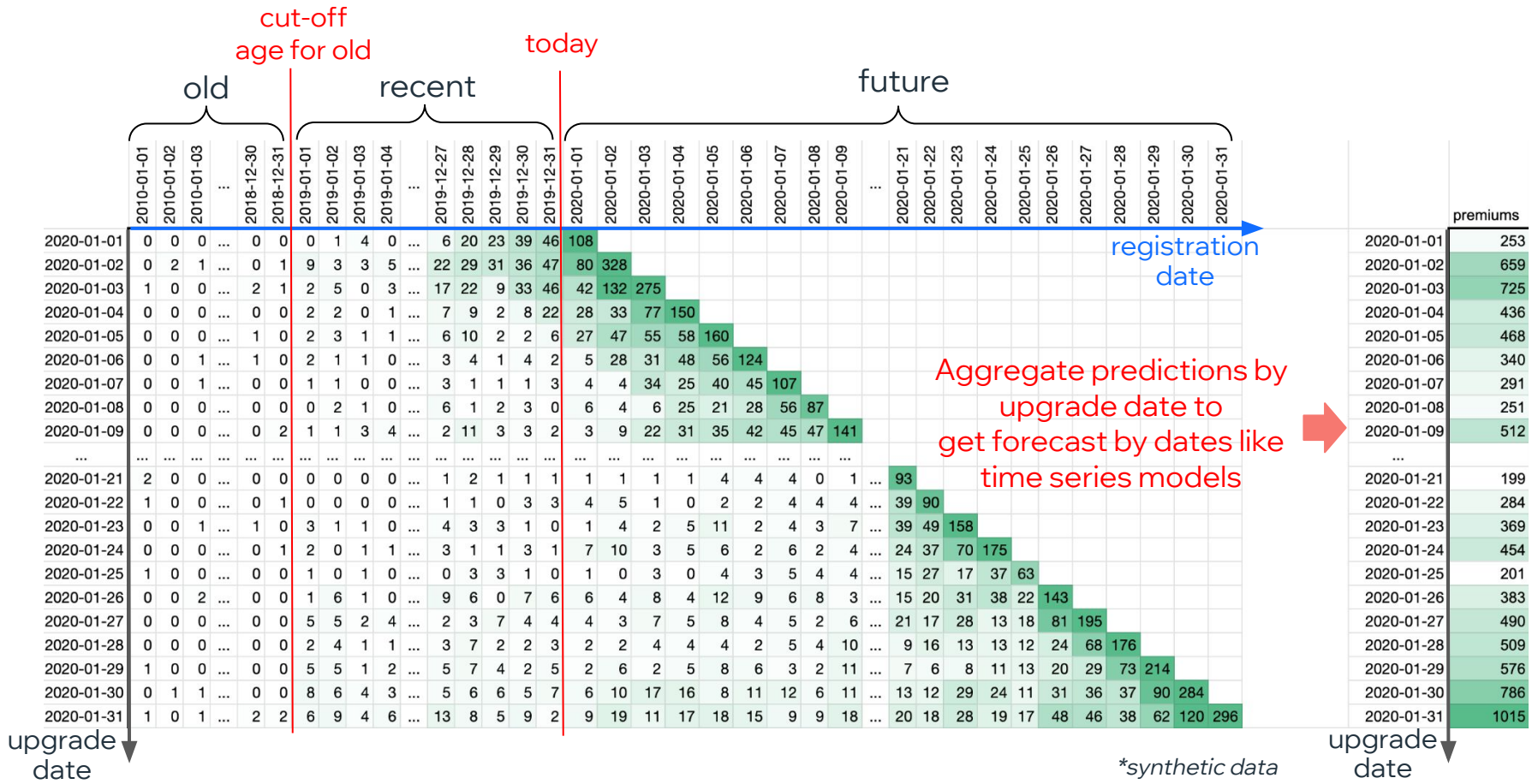
Cohorts in triangle by registration and upgrade date



upgrade date	2010-01-01	2010-01-02	2010-01-03	...	2018-12-30	2018-12-31	2019-01-01	2019-01-02	2019-01-03	2019-01-04	...	2019-12-27	2019-12-28	2019-12-29	2019-12-30	2019-12-31	2020-01-01	2020-01-02	2020-01-03	2020-01-04	2020-01-05	2020-01-06	2020-01-07	2020-01-08	2020-01-09	...	2020-01-21	2020-01-22	2020-01-23	2020-01-24	2020-01-25	2020-01-26	2020-01-27	2020-01-28	2020-01-29	2020-01-30	2020-01-31
2020-01-01	0	0	0	...	0	0	0	1	4	0	...	6	20	23	39	46	108																				
2020-01-02	0	2	1	...	0	1	9	3	3	5	...	22	29	31	36	47	80	328																			
2020-01-03	1	0	0	...	2	1	2	5	0	3	...	17	22	9	33	46	42	132	275																	
2020-01-04	0	0	0	...	0	0	2	2	0	1	...	7	9	2	8	22	28	33	77	150																
2020-01-05	0	0	0	...	1	0	2	3	1	1	...	6	10	2	2	6	27	47	55	58	160															
2020-01-06	0	0	1	...	1	0	2	1	1	0	...	3	4	1	4	2	5	28	31	48	56	124														
2020-01-07	0	0	1	...	0	0	1	1	0	0	...	3	1	1	1	3	4	4	34	25	40	45	107													
2020-01-08	0	0	0	...	0	0	0	2	1	0	...	6	1	2	3	0	6	4	6	25	21	28	56	87												
2020-01-09	0	0	0	...	0	2	1	1	3	4	...	2	11	3	3	2	3	9	22	31	35	42	45	47	141											
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...											
2020-01-21	2	0	0	...	0	0	0	0	0	0	...	1	2	1	1	1	1	1	1	1	4	4	4	0	1	...	93										
2020-01-22	1	0	0	...	0	1	0	0	0	0	...	1	1	0	3	3	4	5	1	0	2	2	4	4	4	...	39	90									
2020-01-23	0	0	1	...	1	0	3	1	1	0	...	4	3	3	1	0	1	4	2	5	11	2	4	3	7	...	39	49	158								
2020-01-24	0	0	0	...	0	1	2	0	1	1	...	3	1	1	3	1	7	10	3	5	6	2	6	2	4	...	24	37	70	175							
2020-01-25	1	0	0	...	0	0	1	0	1	0	...	0	3	3	1	0	1	0	3	0	4	3	5	4	4	...	15	27	17	37	63						
2020-01-26	0	0	2	...	0	0	1	6	1	0	...	9	6	0	7	6	6	4	8	4	12	9	6	8	3	...	15	20	31	38	22	143					
2020-01-27	0	0	0	...	0	0	5	5	2	4	...	2	3	7	4	4	4	3	7	5	8	4	5	2	6	...	21	17	28	13	18	81	195				
2020-01-28	0	0	0	...	0	0	2	4	1	1	...	3	7	2	2	3	2	2	4	4	4	2	5	4	10	...	9	16	13	13	12	24	68	176			
2020-01-29	1	0	0	...	0	0	5	5	1	2	...	5	7	4	2	5	2	6	2	5	8	6	3	2	11	...	7	6	8	11	13	20	29	73	214		
2020-01-30	0	1	1	...	0	0	8	6	4	3	...	5	6	6	5	7	6	10	17	16	8	11	12	6	11	...	13	12	29	24	11	31	36	37	90	284	
2020-01-31	1	0	1	...	2	2	6	9	4	6	...	13	8	5	9	2	9	19	11	17	18	15	9	9	18	...	20	18	28	19	17	48	46	38	62	120	296

*synthetic data

Cohorts in triangle by registration and upgrade date



Regression models

Generalized Linear Model (GLM)

link function

linear predictor

probability distribution

$$g(\mu) = \beta_0 + \beta_1 x_1 + \ldots + \beta_n x_n \quad \text{where: } y \sim P(\mu)$$

$$E(y \mid X) = \mu = g^{-1}(X\beta)$$

- Non-normal distribution of **y** (e.g. from exponential family)
- Linearity between transformed response in terms of the link function and regressors
- Homoscedasticity NOT necessary
- Errors independent, but NOT necessarily normal
- Estimate with MLE instead of OLS

Distribution	Link function
Normal	Identity
Binomial	Logit
Poisson	Log
Gamma	Negative inverse

Regression models

GLM with Poisson distribution and log link function

$$premiums_{(t,r)} \sim Poisson(\lambda_{(t,r)})$$

$$\ln(\lambda_{(t,r)}) = age_{(t,r)} + weekday_{(t)} + dayofyear_{(t)} + holiday_{(t)} + special\_events_{(t,r)} + \ldots$$

where: **t** - upgrade date; **r** - registration date

Pros:

- Supports non-normal distributions from the exponential family (Poisson, Gamma, Tweedie)
- Can extrapolate the trend well

Cons:

- Needs manual feature engineering
- Few python packages have all distributions and regularization
- sensitive to the choice of distribution and link func

Regression models

GLM using statsmodels library in Python

```
from statsmodels.genmod.generalized_linear_model import GLM
from statsmodels.genmod.families import Poisson

formula = 'premiums ~
                      log1p_age + f1_age + ... + f10_age +
                      f1_numdate + ... + f6_numdate
                      f1_weekday + ... + f5_weekday +
                      f1_dayofyear + ... + f10_dayofyear +
                      holiday +
                      f1_economy_change + ... + f1_fx_rate +
                      special_event + f1_special_event_days + f2_special_event_days +
                      new_version + package_update +
                      log1p_age*new_version + log1p_age*special_event'

model = GLM.from_formula(formula=formula, data=df_train, family=Poisson())
model_fit = model.fit()
```

Regression models

GLM using statsmodels library in Python

```
from statsmodels.genmod.generalized_linear_model import GLM
from statsmodels.genmod.families import Poisson

formula = 'premiums ~
        log1p_age + f1_age + ... + f10_age +
        f1_numdate + ... + f6_numdate
        f1_weekday + ... + f5_weekday +
        f1_dayofyear + ... + f10_dayofyear +
        holiday +
        f1_economy_change + ... + f1_fx_rate +
        special_event + f1_special_event_days + f2_special_event_days +
        new_version + package_update +
        log1p_age*new_version + log1p_age*special_event'

model = GLM.from_formula(formula=formula, data=df_train, family=Poisson())
model_fit = model.fit()
```

decay by age

time-series decomp: trend, seasonality, holidays

external factors

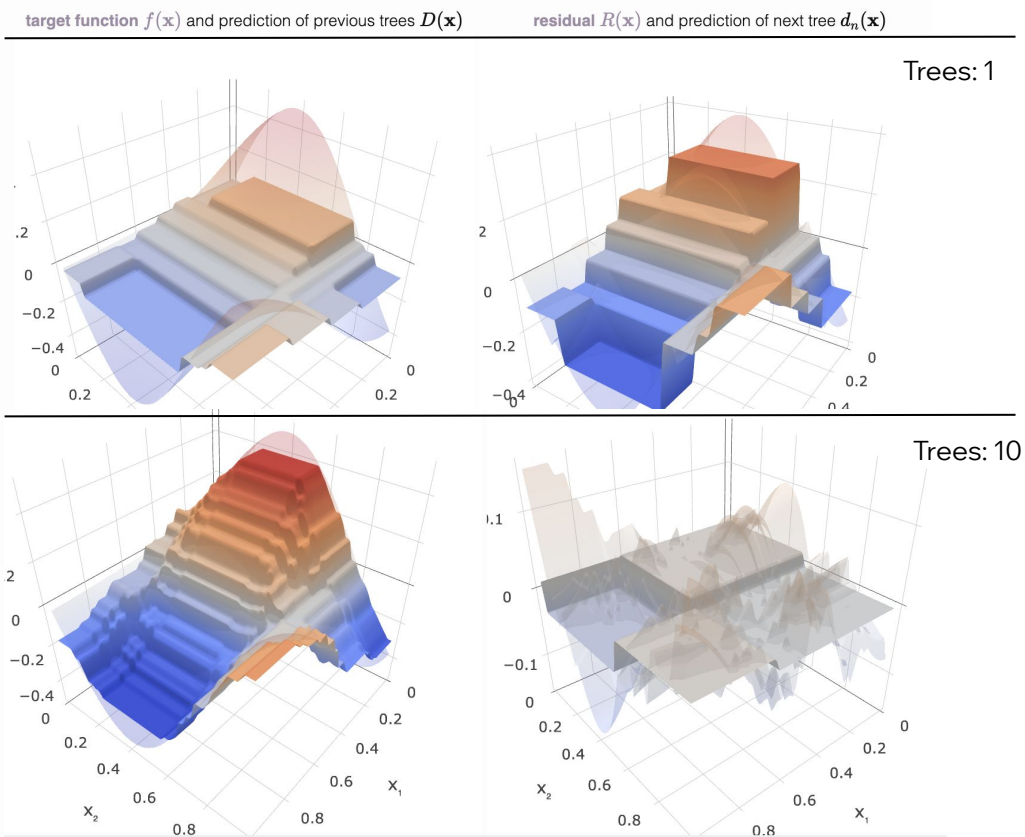
business

interaction terms

Note: **f1_, f2_ ...** prefixes mean that this feature is a transformation of the raw feature, for example with a spline function

Regression models

Gradient Boosted Machine



How it works:

- Each tree is trained on residuals from the previous iteration, taken with a small rate
- Prediction is reconstructed from the residuals predicted by all trees

* pictures from: https://arogozhnikov.github.io/

Regression models

Gradient Boosted Machine

Pros:

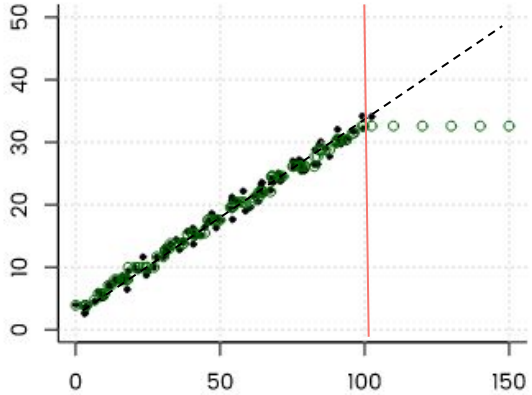
- Handles non-linearities and interactions, none or little feature engineering is needed

- Supports non-normal distributions from the exponential family (Poisson, Gamma, Tweedie)

- Less sensitive to the choice of distribution

Cons:

- Can't extrapolate on unseen data, e.g. can't extrapolate the trend.



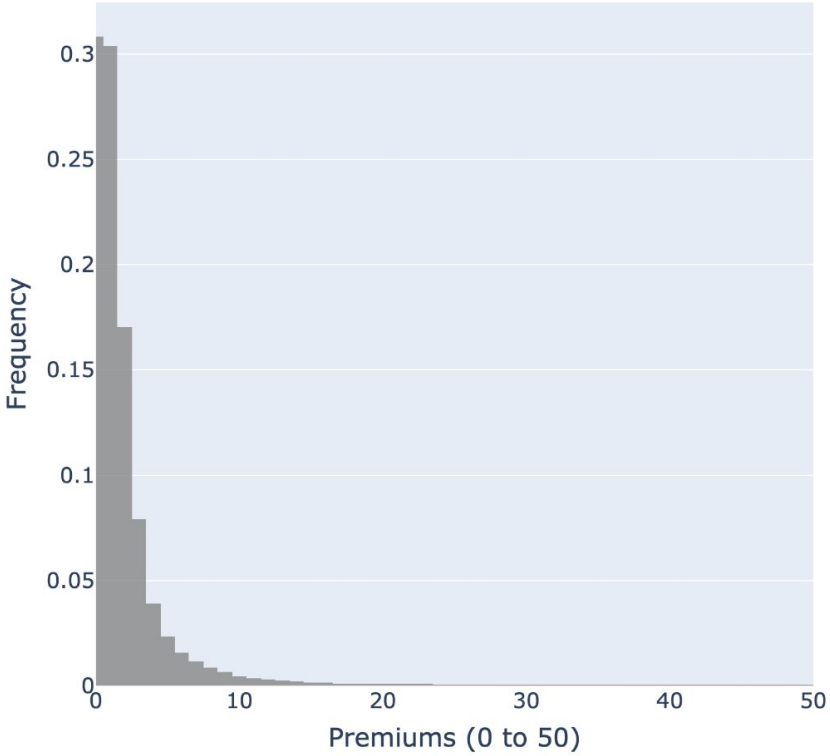
Regression models

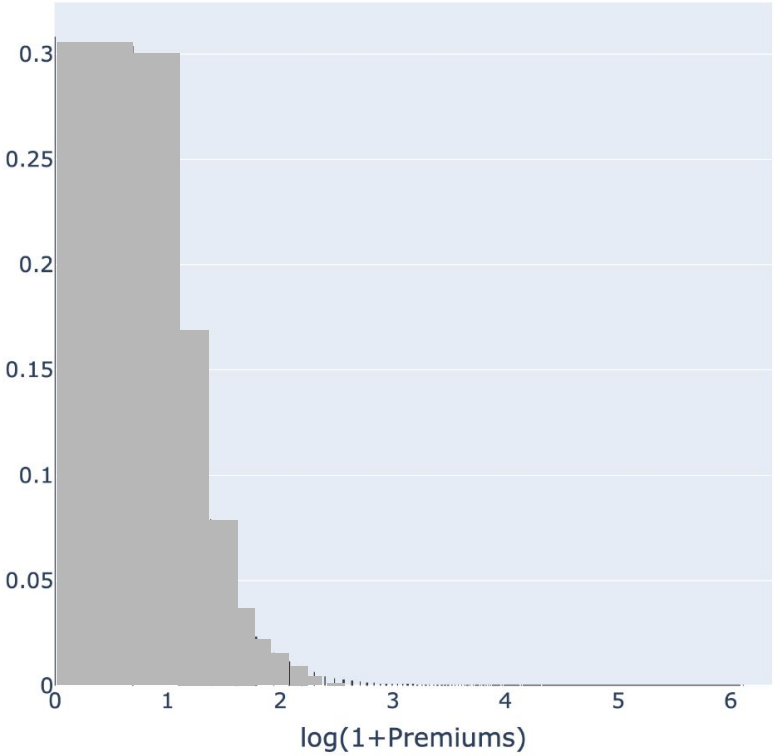
Gradient Boosted Machine (GBM) using lightgbm in Python

```
import lightgbm
features = ['age', 'numdate', 'weekday', 'dayofyear', 'economy_change',
            'fx_rate', 'special_event', 'new_version', 'package_update']
params = {
    'objective': 'poisson',
    'boosting': 'gbdt',
    'max_depth': 4,
    'num_leaves': 15,
    'min_data_in_leaf': 200,
    'feature_fraction': 0.50,
    'bagging_fraction': 0.50,
    'learning_rate': 0.15
}
model = lightgbm.train(
    params=params, train_set=df_train, valid_sets=[df_valid, df_train],
    num_boost_round=1000,
    early_stopping_rounds=100
)
```

Choosing the right distribution

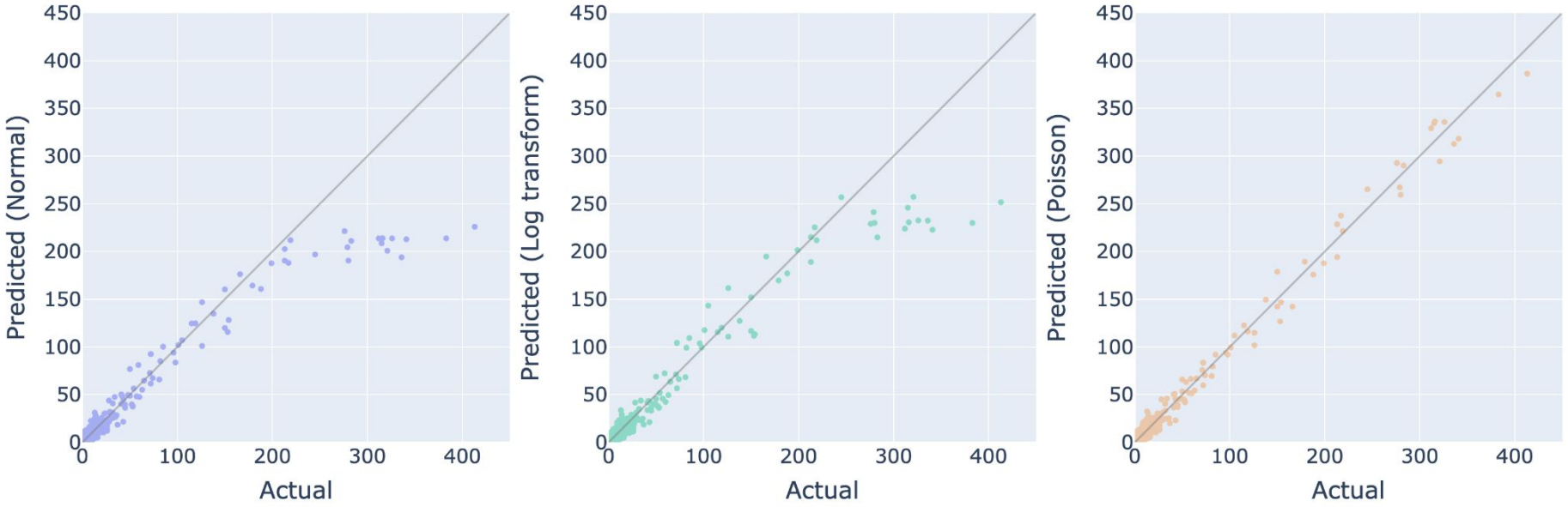
Target is non-normal, highly skewed, exponential-like





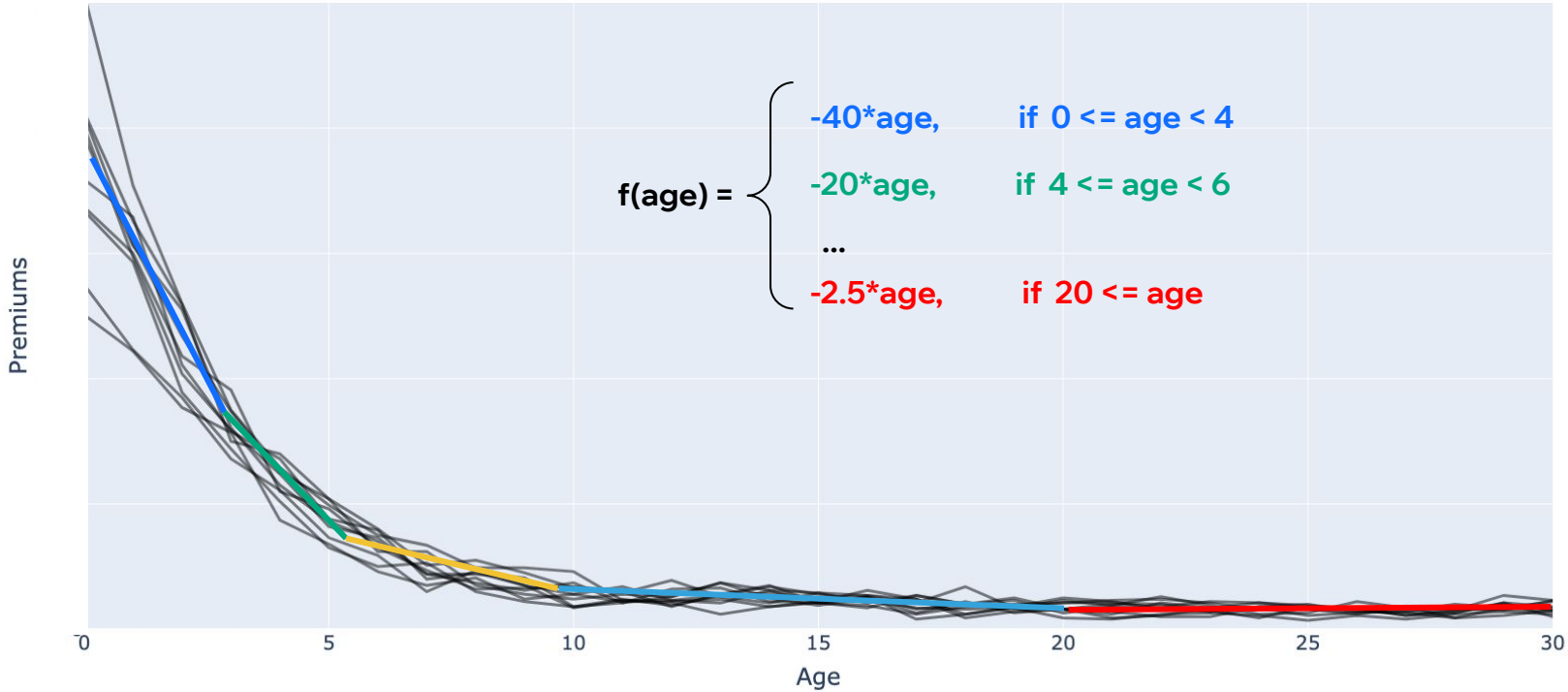
Choosing the right distribution

Normal, Log-transformed, Poisson



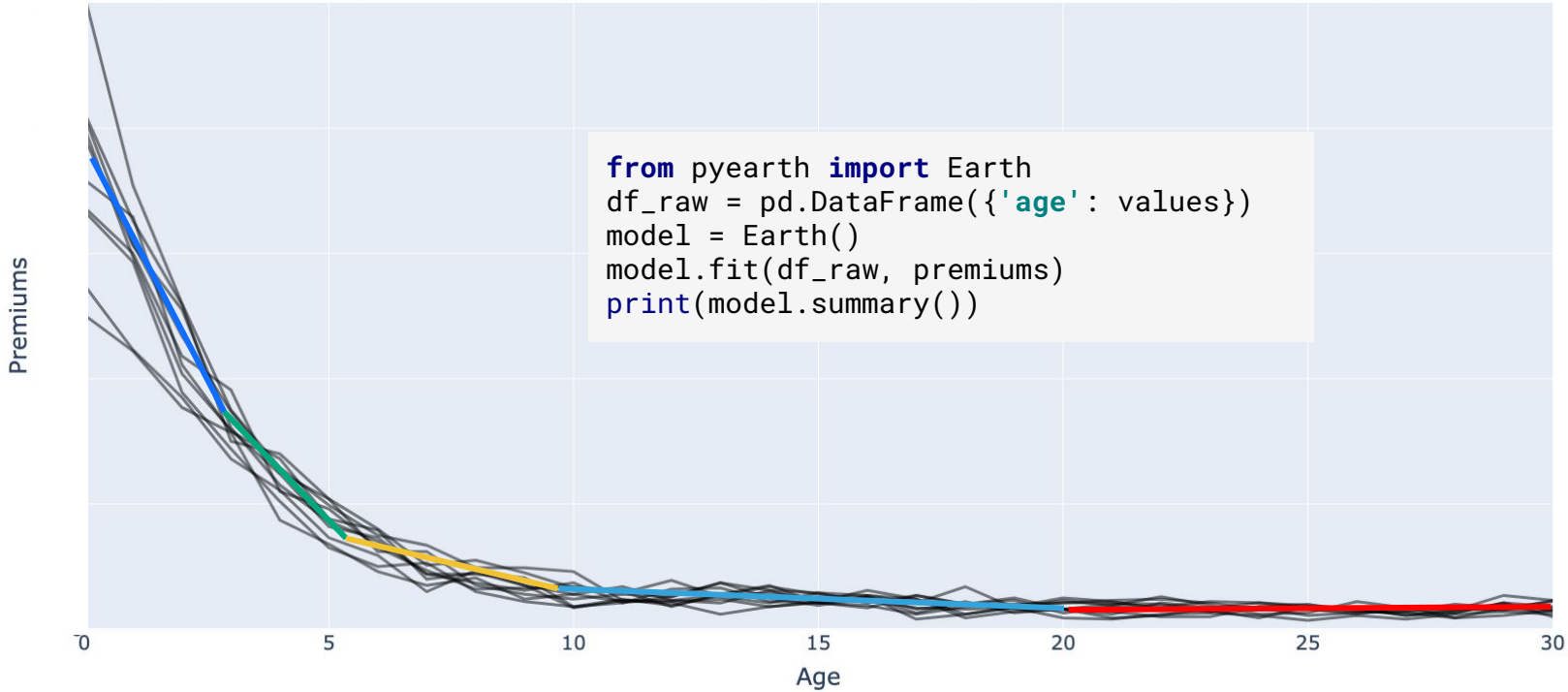
Modelling the exponential decay by age

Piecewise linear function



Modelling the exponential decay by age

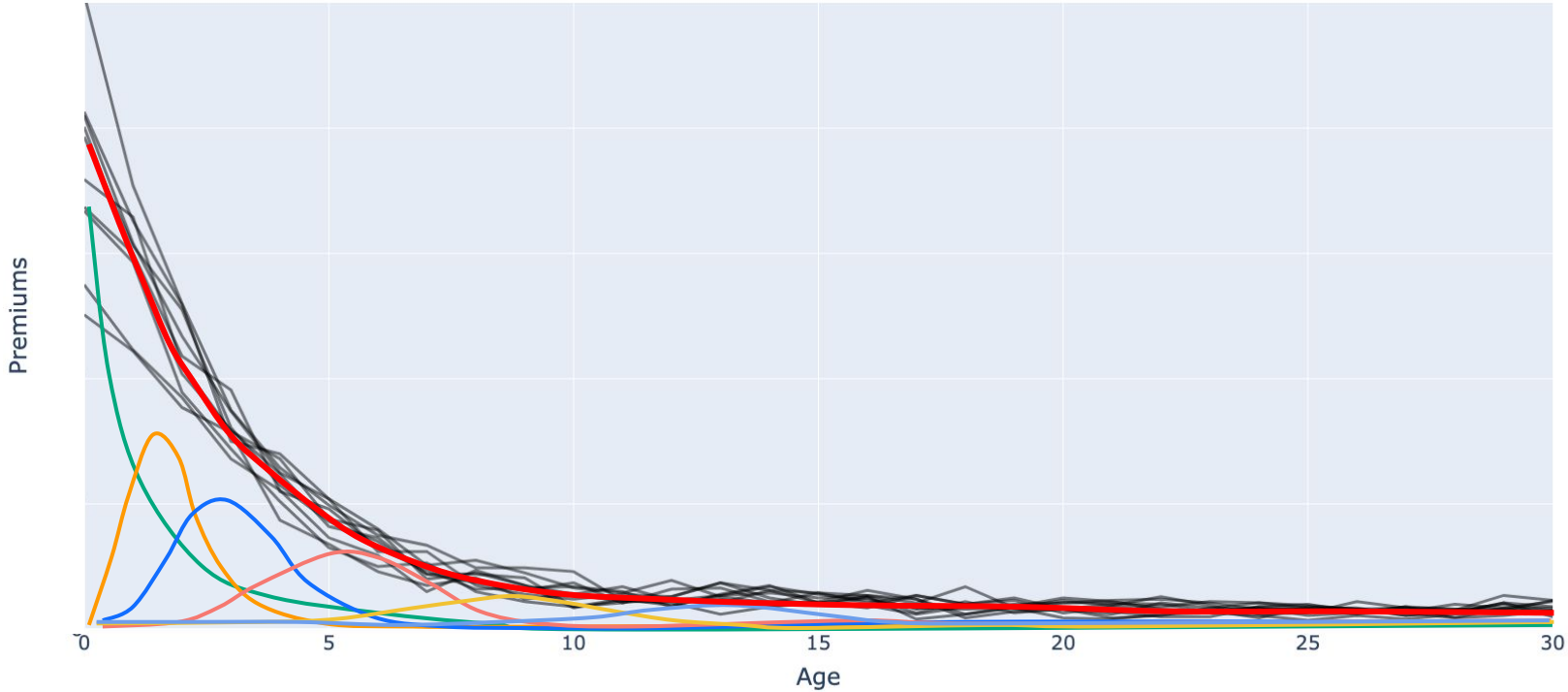
find knots with MARS from *pyearth* (python), *earth* (R)




```python
from pyearth import Earth
df_raw = pd.DataFrame({'age': values})
model = Earth()
model.fit(df_raw, premiums)
print(model.summary())
```

Modelling the exponential decay by age

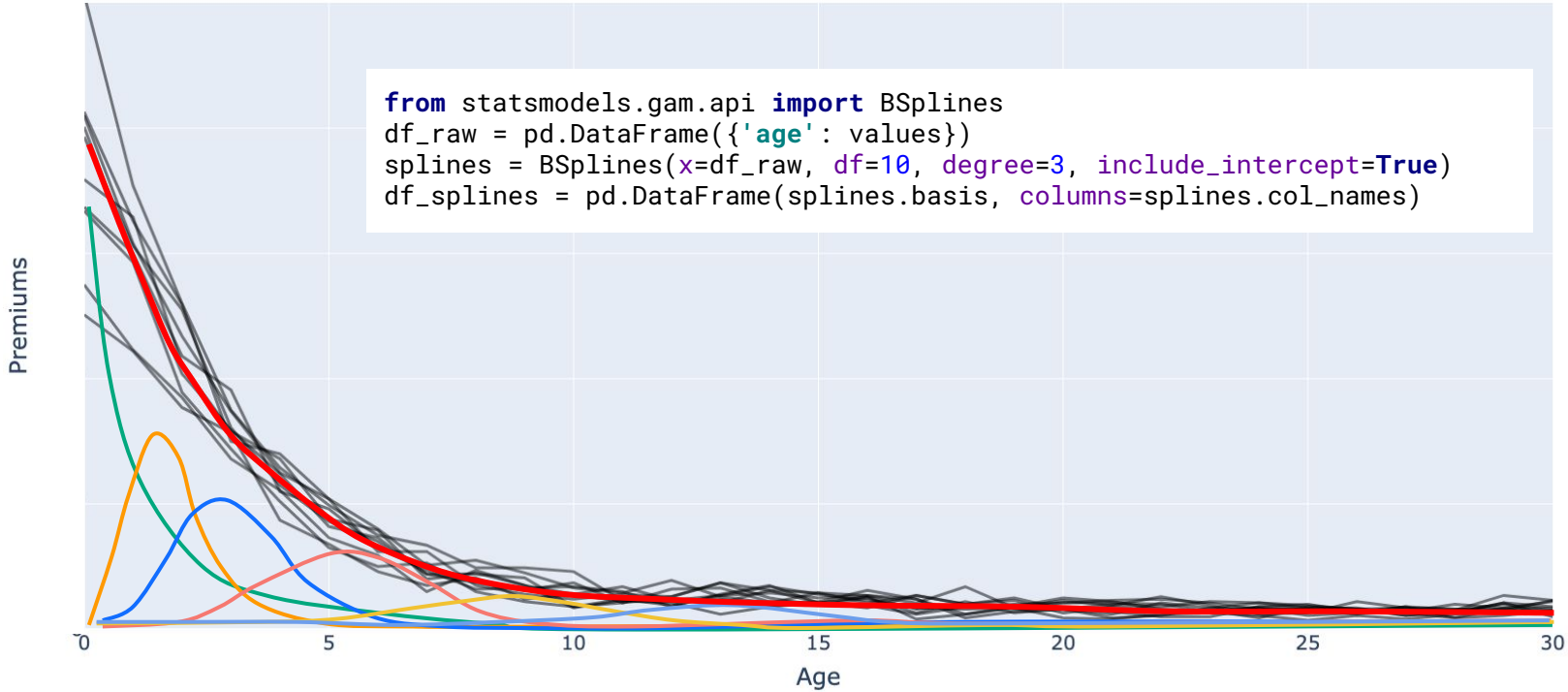
B-splines smoothing



Modelling the exponential decay by age

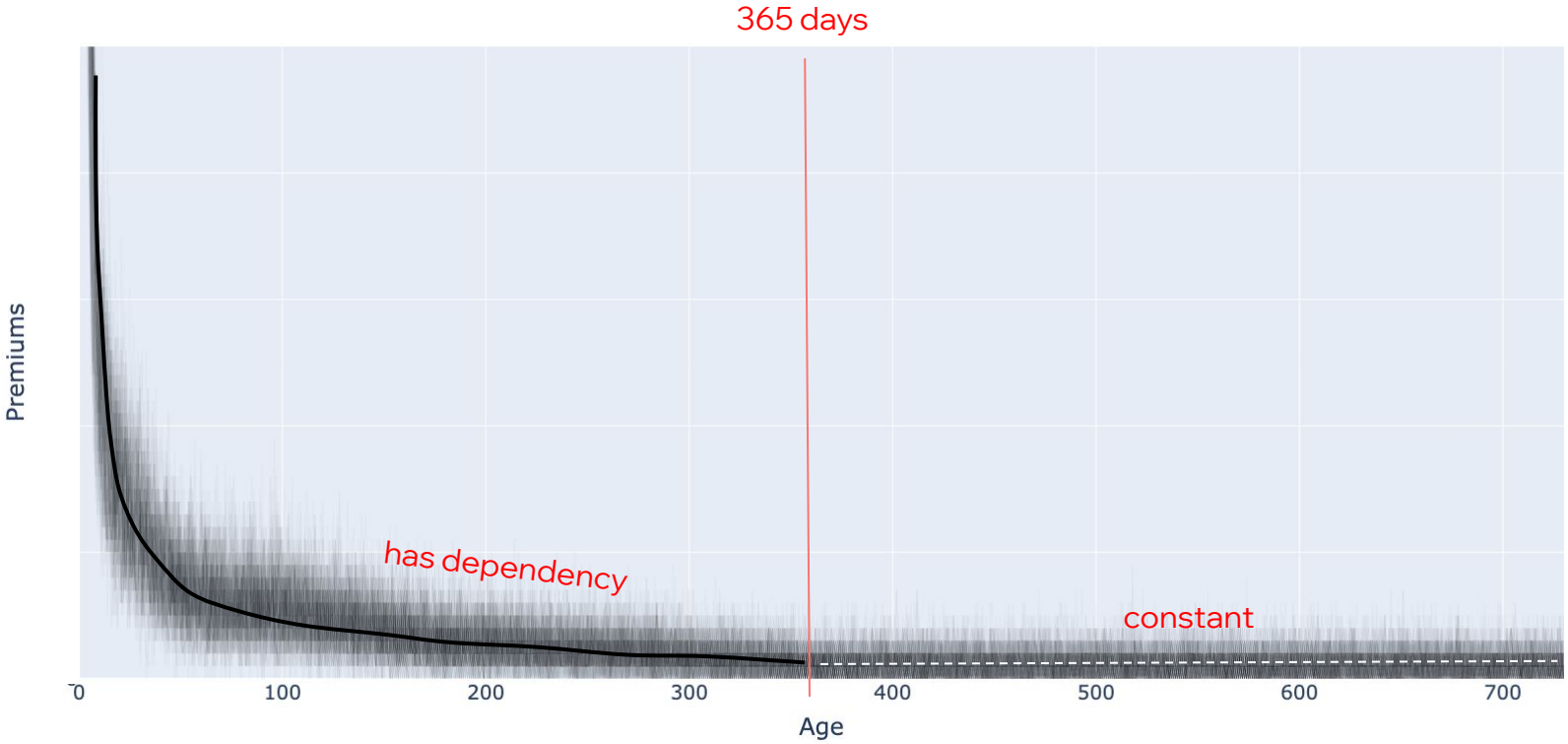
B-splines smoothing using *statsmodels* (Python)

```
from statsmodels.gam.api import BSplines
df_raw = pd.DataFrame({'age': values})
splines = BSplines(x=df_raw, df=10, degree=3, include_intercept=True)
df_splines = pd.DataFrame(splines.basis, columns=splines.col_names)
```



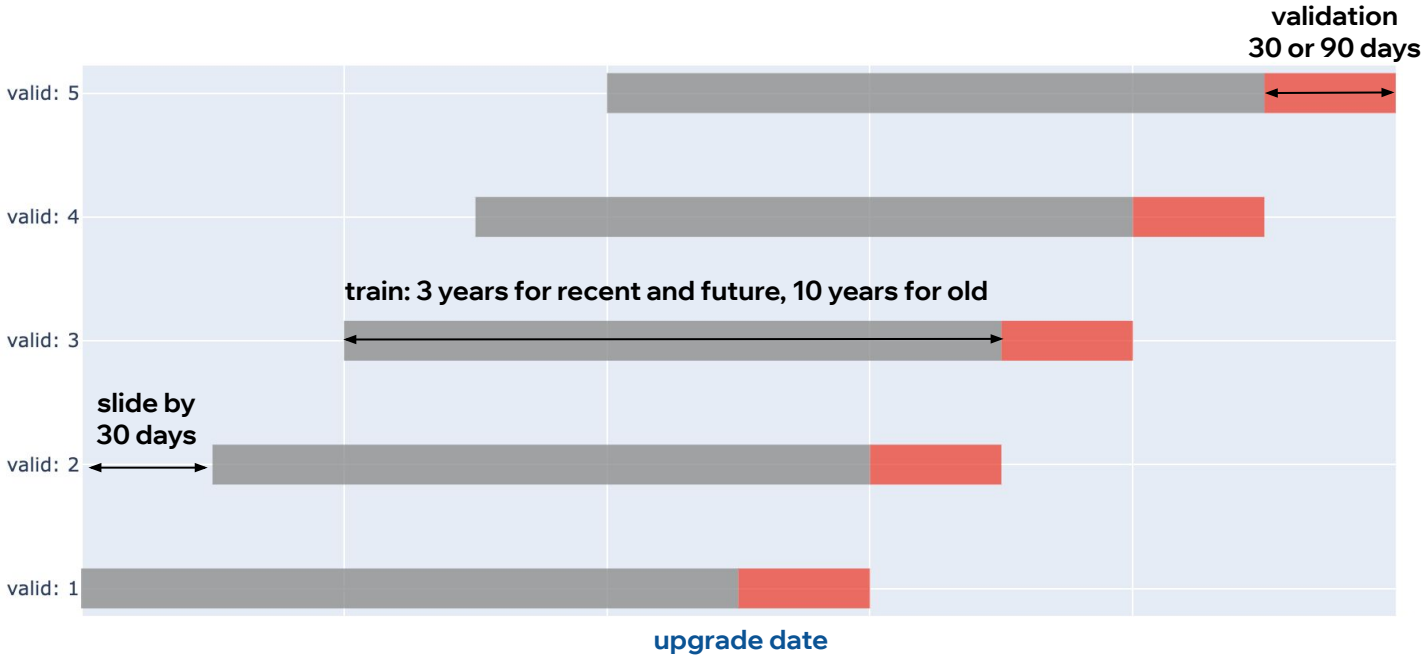
Cut-off between old and recent cohorts

where premiums become constant, independent of age



Validation

rolling walk forward by upgrade date



Models comparison on walk-fwd validation

errors of forecasts for 30 days ahead

Model	*MAPE%*
Ensemble:	2.99%
Cohort GBM:	3.03%
Prophet:	3.78%
Cohort GLM:	3.96%

** this is an actual print-screen from our internal reports*

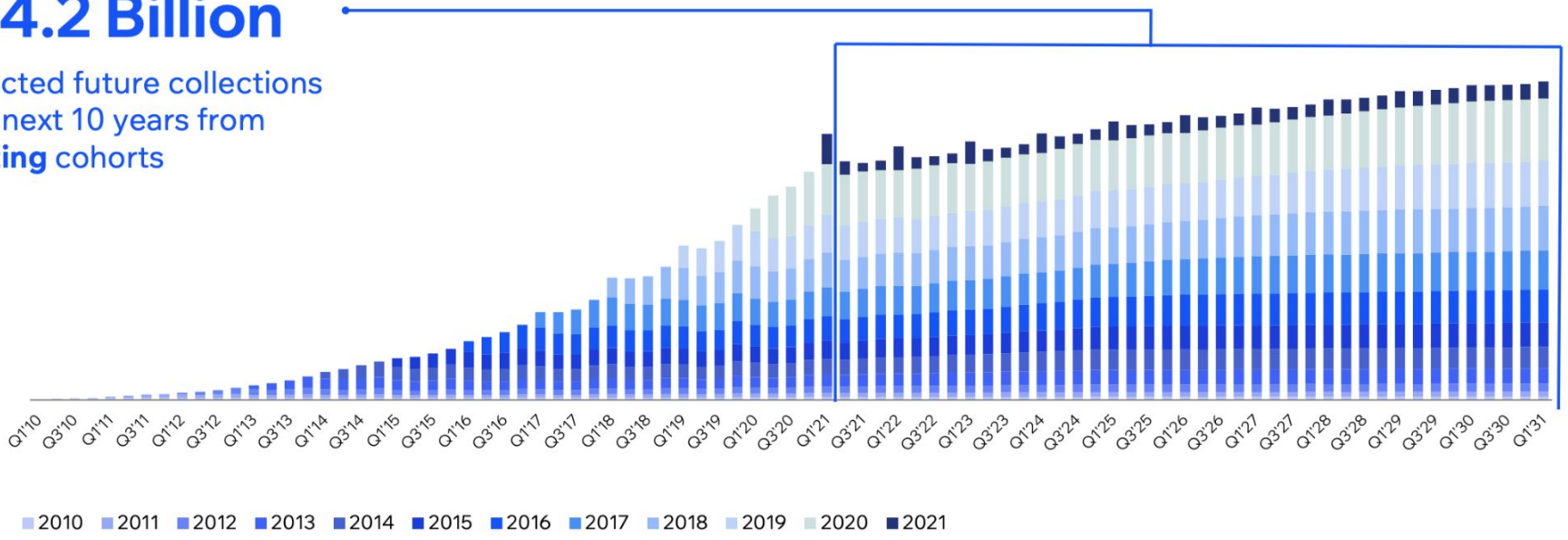
Bonus:

Using cohort-based models you can **aggregate** predictions **by registration date** too.
This allows to understand **which cohorts** generate **how much** out of the total forecast.

Cohort based models
can tell which cohorts will generate premiums

$14.2 Billion

Expected future collections over next 10 years from **existing** cohorts



** as of 2021-03-31, from investor presentation, find more: https://investors.wix.com/*

That’s it.
Thank you!

Contact me:

nicolaiv@wix.com
https://www.linkedin.com/in/nicolai-vicol/

Blog about engineering & data science at Wix:

https://medium.com/wix-engineering

Questions?
There are no stupid questions.
So, please ask :)

Question:

How data was generated?

Answer:

https://github.com/nicolaivico/gen-synth-data-cohorts/blob/main/gen_data.py

```
# - multiplicative model for users:
# users(r) = trend(r) * yearly(r) * weekly(r) * holiday(r)
# where: r - registration date, t - upgrade date

# - model for the exponential decay of premiums by age:
# premium_rate(age) = f(age)
# where: f(age) is a non-linear function

# - Poisson model for premiums:
# lambda(r,t) = offers(t) * trend(r) * yearly(r) * weekly(r) * premium_rate(age=t-r) * frees(r)
# premiums(r,t) ~ Poisson(lambda(r,t))
```

Question:

How to deal with the extrapolation issue by GBMs?

Answer:

I don’t have a good answer.
But maybe a simple and stupid solution is to perform the cross validation and check the bias in errors.
If any consistent bias of the same sign, adjust forecast by the percent of bias with opposite sign.

Appendix

Time-series models

ARIMA(p, d, q)

Auto**R**egressive **I**ntegrated **M**oving **A**verage, e.g. ARIMA(1, 1, 1)

differenced target series		autoregressive part		moving average part		error term
$\Delta y_{(t)}$	=	$\varphi \Delta y_{(t-1)}$	+	$\theta \varepsilon_{(t-1)}$	+	$\varepsilon_{(t)}$

p - autoregressive (AR) order
d - order of differencing
q - moving average (MA) order

Time-series models

SARIMAX(p, d, q)(P, D, Q, s)

Seasonal **A**uto**R**egressive **I**ntegrated **M**oving **A**verage with e**X**ogenous regressors

e.g. SARIMAX(1, 1, 1)(1, 1, 1, 12)

$$\Delta y_{(t)} = \ldots + \varphi(\Delta y_{(t-12)} + \Delta y_{(t-13)})$$

seasonal autoregressive part

$$+ \theta(\varepsilon_{(t-12)} + \varepsilon_{(t-13)})$$

seasonal moving-average part

$$+ \beta X_{(t)}$$

exogenous variables

P, **D**, **Q** - seasonal orders

s - seasonal cycle

Time-series models

Prophet, STL

target series: $y(t)$ = trend, "growth" term: $g(t)$ + seasonal terms: $s(t)$ + holidays & special events: $h(t)$ + error term: ε

where: **t** - date (time index)

SEATS, X11

"Industry standard" used by national banks & government agencies for macroeconomic data having monthly or quarterly frequency.

LSTM

Long **S**hort-**T**erm **M**emory networks are capable of capturing patterns in the time series through sequence to sequence training and generate future sequences, i.e. forecasts.