

# Métricas de Qualidade de Código para Projetos de Ciência de Dados e Aprendizado de Máquina

Yesman Choque Mamani Orientador: Prof. Dr. Fabio Kon

Instituto de Matemática, Estatística e Ciência da Computação – Universidade de São Paulo (IME-USP)

## Introdução

A popularização de DS/ML trouxe desafios de qualidade de código. O desenvolvimento por não-especialistas em Engenharia de Software gera problemas de manutenibilidade e legibilidade.

### Objetivos:

- Catalogar antipadrões de DS/ML.
- Desenvolver o **DeSmell** para detecção.
- Validar com usuários.

## O que é Code Smell?

Sintomas no código que indicam possíveis problemas de projeto. Não são bugs, mas dificultam a manutenção e evolução do software.

## Catálogo de Antipadrões

**Metodologia:** Análise de 3 projetos (EconML, FinRL, MONAI) com Pylint.

### Top 5 Antipadrões:

1. Alta Complexidade Ciclomática
2. Iterações Desnecessárias
3. Código Duplicado
4. Nomes Pouco Claros
5. Muitos Argumentos

## Exemplo de Antipadrão

Detecção e visualização de code smells no código fonte.

```
Occurrences: env_stocktrading_cashpenalty.py ↔ env_stocktrading_stoploss.py
```

```
finrl/meta/env_stock_trad...
19 class StockTradingEnvCashpenalty(gym.Env):
262     def get_transactions(self, actions):
278         if self.discrete_actions:
279             # convert into integer because we can't buy
280             actions = actions // self.closings
281             # round down actions to the nearest multiple
282             actions = np.where(actions >= 0,
283                 (actions // self.shares_increment) * self.s
284                 * self.shares_increment,
285             )
286             else:
287                 actions = actions / self.closings
288
289             # can't sell more than we have
290             actions = np.maximum(actions, -np.array(self.
291
292             # deal with turbulence
293             actions = np.array(actions, dtype=np.int32)
294
295             # convert into integer because we can't buy
296             actions = actions // self.closings
297             # round down actions to the nearest multiple
298             actions = np.where(actions >= 0,
299                 (actions // self.shares_increment) * self.s
300                 * self.shares_increment,
301             )
302             else:
303                 actions = np.where(actions > 0, actions
304
305             # clip actions so we can't sell more assets than we have
306             actions = np.minimum(actions, -np.array(self.
307
308             # deal with turbulence
309             actions = np.array(actions, dtype=np.int32)
310
311             # convert into integer because we can't buy
312             actions = actions // self.closings
313             # round down actions to the nearest multiple
314             actions = np.where(actions >= 0,
315                 (actions // self.shares_increment) * self.s
316                 * self.shares_increment,
317             )
318             else:
319                 actions = np.where(actions > 0, actions
320
321             # clip actions so we can't sell more assets than we have
322             actions = np.minimum(actions, -np.array(self.
323
324             # deal with turbulence
325             actions = np.array(actions, dtype=np.int32)
326
327             # convert into integer because we can't buy
328             actions = actions // self.closings
329             # round down actions to the nearest multiple
330             actions = np.where(actions >= 0,
331                 (actions // self.shares_increment) * self.s
332                 * self.shares_increment,
333             )
334             else:
335                 actions = np.where(actions > 0, actions
336
337             # clip actions so we can't sell more assets than we have
338             actions = np.minimum(actions, -np.array(self.
339
340             # deal with turbulence
341             actions = np.array(actions, dtype=np.int32)
342
343             # convert into integer because we can't buy
344             actions = actions // self.closings
345             # round down actions to the nearest multiple
346             actions = np.where(actions >= 0,
347                 (actions // self.shares_increment) * self.s
348                 * self.shares_increment,
349             )
350             else:
351                 actions = np.where(actions > 0, actions
352
353             # clip actions so we can't sell more assets than we have
354             actions = np.minimum(actions, -np.array(self.
355
356             # deal with turbulence
357             actions = np.array(actions, dtype=np.int32)
358
359             # convert into integer because we can't buy
360             actions = actions // self.closings
361             # round down actions to the nearest multiple
362             actions = np.where(actions >= 0,
363                 (actions // self.shares_increment) * self.s
364                 * self.shares_increment,
365             )
366             else:
367                 actions = np.where(actions > 0, actions
368
369             # clip actions so we can't sell more assets than we have
370             actions = np.minimum(actions, -np.array(self.
371
372             # deal with turbulence
373             actions = np.array(actions, dtype=np.int32)
374
375             # convert into integer because we can't buy
376             actions = actions // self.closings
377             # round down actions to the nearest multiple
378             actions = np.where(actions >= 0,
379                 (actions // self.shares_increment) * self.s
380                 * self.shares_increment,
381             )
382             else:
383                 actions = np.where(actions > 0, actions
384
385             # clip actions so we can't sell more assets than we have
386             actions = np.minimum(actions, -np.array(self.
387
388             # deal with turbulence
389             actions = np.array(actions, dtype=np.int32)
390
391             # convert into integer because we can't buy
392             actions = actions // self.closings
393             # round down actions to the nearest multiple
394             actions = np.where(actions >= 0,
395                 (actions // self.shares_increment) * self.s
396                 * self.shares_increment,
397             )
398             else:
399                 actions = np.where(actions > 0, actions
400
401             # clip actions so we can't sell more assets than we have
402             actions = np.minimum(actions, -np.array(self.
403
404             # deal with turbulence
405             actions = np.array(actions, dtype=np.int32)
406
407             # convert into integer because we can't buy
408             actions = actions // self.closings
409             # round down actions to the nearest multiple
410             actions = np.where(actions >= 0,
411                 (actions // self.shares_increment) * self.s
412                 * self.shares_increment,
413             )
414             else:
415                 actions = np.where(actions > 0, actions
416
417             # clip actions so we can't sell more assets than we have
418             actions = np.minimum(actions, -np.array(self.
419
420             # deal with turbulence
421             actions = np.array(actions, dtype=np.int32)
422
423             # convert into integer because we can't buy
424             actions = actions // self.closings
425             # round down actions to the nearest multiple
426             actions = np.where(actions >= 0,
427                 (actions // self.shares_increment) * self.s
428                 * self.shares_increment,
429             )
430             else:
431                 actions = np.where(actions > 0, actions
432
433             # clip actions so we can't sell more assets than we have
434             actions = np.minimum(actions, -np.array(self.
435
436             # deal with turbulence
437             actions = np.array(actions, dtype=np.int32)
438
439             # convert into integer because we can't buy
440             actions = actions // self.closings
441             # round down actions to the nearest multiple
442             actions = np.where(actions >= 0,
443                 (actions // self.shares_increment) * self.s
444                 * self.shares_increment,
445             )
446             else:
447                 actions = np.where(actions > 0, actions
448
449             # clip actions so we can't sell more assets than we have
450             actions = np.minimum(actions, -np.array(self.
451
452             # deal with turbulence
453             actions = np.array(actions, dtype=np.int32)
454
455             # convert into integer because we can't buy
456             actions = actions // self.closings
457             # round down actions to the nearest multiple
458             actions = np.where(actions >= 0,
459                 (actions // self.shares_increment) * self.s
460                 * self.shares_increment,
461             )
462             else:
463                 actions = np.where(actions > 0, actions
464
465             # clip actions so we can't sell more assets than we have
466             actions = np.minimum(actions, -np.array(self.
467
468             # deal with turbulence
469             actions = np.array(actions, dtype=np.int32)
470
471             # convert into integer because we can't buy
472             actions = actions // self.closings
473             # round down actions to the nearest multiple
474             actions = np.where(actions >= 0,
475                 (actions // self.shares_increment) * self.s
476                 * self.shares_increment,
477             )
478             else:
479                 actions = np.where(actions > 0, actions
480
481             # clip actions so we can't sell more assets than we have
482             actions = np.minimum(actions, -np.array(self.
483
484             # deal with turbulence
485             actions = np.array(actions, dtype=np.int32)
486
487             # convert into integer because we can't buy
488             actions = actions // self.closings
489             # round down actions to the nearest multiple
490             actions = np.where(actions >= 0,
491                 (actions // self.shares_increment) * self.s
492                 * self.shares_increment,
493             )
494             else:
495                 actions = np.where(actions > 0, actions
496
497             # clip actions so we can't sell more assets than we have
498             actions = np.minimum(actions, -np.array(self.
499
499             # deal with turbulence
500             actions = np.array(actions, dtype=np.int32)
501
502             # convert into integer because we can't buy
503             actions = actions // self.closings
504             # round down actions to the nearest multiple
505             actions = np.where(actions >= 0,
506                 (actions // self.shares_increment) * self.s
507                 * self.shares_increment,
508             )
509             else:
510                 actions = np.where(actions > 0, actions
511
511             # clip actions so we can't sell more assets than we have
512             actions = np.minimum(actions, -np.array(self.
513
513             # deal with turbulence
514             actions = np.array(actions, dtype=np.int32)
515
516             # convert into integer because we can't buy
517             actions = actions // self.closings
518             # round down actions to the nearest multiple
519             actions = np.where(actions >= 0,
520                 (actions // self.shares_increment) * self.s
521                 * self.shares_increment,
522             )
523             else:
524                 actions = np.where(actions > 0, actions
525
525             # clip actions so we can't sell more assets than we have
526             actions = np.minimum(actions, -np.array(self.
527
527             # deal with turbulence
528             actions = np.array(actions, dtype=np.int32)
529
530             # convert into integer because we can't buy
531             actions = actions // self.closings
532             # round down actions to the nearest multiple
533             actions = np.where(actions >= 0,
534                 (actions // self.shares_increment) * self.s
535                 * self.shares_increment,
536             )
537             else:
538                 actions = np.where(actions > 0, actions
539
539             # clip actions so we can't sell more assets than we have
540             actions = np.minimum(actions, -np.array(self.
541
541             # deal with turbulence
542             actions = np.array(actions, dtype=np.int32)
543
544             # convert into integer because we can't buy
545             actions = actions // self.closings
546             # round down actions to the nearest multiple
547             actions = np.where(actions >= 0,
548                 (actions // self.shares_increment) * self.s
549                 * self.shares_increment,
550             )
551             else:
552                 actions = np.where(actions > 0, actions
553
553             # clip actions so we can't sell more assets than we have
554             actions = np.minimum(actions, -np.array(self.
555
555             # deal with turbulence
556             actions = np.array(actions, dtype=np.int32)
557
558             # convert into integer because we can't buy
559             actions = actions // self.closings
560             # round down actions to the nearest multiple
561             actions = np.where(actions >= 0,
562                 (actions // self.shares_increment) * self.s
563                 * self.shares_increment,
564             )
565             else:
566                 actions = np.where(actions > 0, actions
567
567             # clip actions so we can't sell more assets than we have
568             actions = np.minimum(actions, -np.array(self.
569
569             # deal with turbulence
570             actions = np.array(actions, dtype=np.int32)
571
572             # convert into integer because we can't buy
573             actions = actions // self.closings
574             # round down actions to the nearest multiple
575             actions = np.where(actions >= 0,
576                 (actions // self.shares_increment) * self.s
577                 * self.shares_increment,
578             )
579             else:
580                 actions = np.where(actions > 0, actions
581
581             # clip actions so we can't sell more assets than we have
582             actions = np.minimum(actions, -np.array(self.
583
583             # deal with turbulence
584             actions = np.array(actions, dtype=np.int32)
585
586             # convert into integer because we can't buy
587             actions = actions // self.closings
588             # round down actions to the nearest multiple
589             actions = np.where(actions >= 0,
590                 (actions // self.shares_increment) * self.s
591                 * self.shares_increment,
592             )
593             else:
594                 actions = np.where(actions > 0, actions
595
595             # clip actions so we can't sell more assets than we have
596             actions = np.minimum(actions, -np.array(self.
597
597             # deal with turbulence
598             actions = np.array(actions, dtype=np.int32)
599
599             # convert into integer because we can't buy
600             actions = actions // self.closings
601             # round down actions to the nearest multiple
602             actions = np.where(actions >= 0,
603                 (actions // self.shares_increment) * self.s
604                 * self.shares_increment,
605             )
606             else:
607                 actions = np.where(actions > 0, actions
608
608             # clip actions so we can't sell more assets than we have
609             actions = np.minimum(actions, -np.array(self.
610
610             # deal with turbulence
611             actions = np.array(actions, dtype=np.int32)
612
612             # convert into integer because we can't buy
613             actions = actions // self.closings
614             # round down actions to the nearest multiple
615             actions = np.where(actions >= 0,
616                 (actions // self.shares_increment) * self.s
617                 * self.shares_increment,
618             )
619             else:
620                 actions = np.where(actions > 0, actions
621
621             # clip actions so we can't sell more assets than we have
622             actions = np.minimum(actions, -np.array(self.
623
623             # deal with turbulence
624             actions = np.array(actions, dtype=np.int32)
625
625             # convert into integer because we can't buy
626             actions = actions // self.closings
627             # round down actions to the nearest multiple
628             actions = np.where(actions >= 0,
629                 (actions // self.shares_increment) * self.s
630                 * self.shares_increment,
631             )
632             else:
633                 actions = np.where(actions > 0, actions
634
634             # clip actions so we can't sell more assets than we have
635             actions = np.minimum(actions, -np.array(self.
636
636             # deal with turbulence
637             actions = np.array(actions, dtype=np.int32)
638
638             # convert into integer because we can't buy
639             actions = actions // self.closings
640             # round down actions to the nearest multiple
641             actions = np.where(actions >= 0,
642                 (actions // self.shares_increment) * self.s
643                 * self.shares_increment,
644             )
645             else:
646                 actions = np.where(actions > 0, actions
647
647             # clip actions so we can't sell more assets than we have
648             actions = np.minimum(actions, -np.array(self.
649
649             # deal with turbulence
650             actions = np.array(actions, dtype=np.int32)
651
651             # convert into integer because we can't buy
652             actions = actions // self.closings
653             # round down actions to the nearest multiple
654             actions = np.where(actions >= 0,
655                 (actions // self.shares_increment) * self.s
656                 * self.shares_increment,
657             )
658             else:
659                 actions = np.where(actions > 0, actions
660
660             # clip actions so we can't sell more assets than we have
661             actions = np.minimum(actions, -np.array(self.
662
662             # deal with turbulence
663             actions = np.array(actions, dtype=np.int32)
664
664             # convert into integer because we can't buy
665             actions = actions // self.closings
666             # round down actions to the nearest multiple
667             actions = np.where(actions >= 0,
668                 (actions // self.shares_increment) * self.s
669                 * self.shares_increment,
670             )
671             else:
672                 actions = np.where(actions > 0, actions
673
673             # clip actions so we can't sell more assets than we have
674             actions = np.minimum(actions, -np.array(self.
675
675             # deal with turbulence
676             actions = np.array(actions, dtype=np.int32)
677
677             # convert into integer because we can't buy
678             actions = actions // self.closings
679             # round down actions to the nearest multiple
680             actions = np.where(actions >= 0,
681                 (actions // self.shares_increment) * self.s
682                 * self.shares_increment,
683             )
684             else:
685                 actions = np.where(actions > 0, actions
686
686             # clip actions so we can't sell more assets than we have
687             actions = np.minimum(actions, -np.array(self.
688
688             # deal with turbulence
689             actions = np.array(actions, dtype=np.int32)
690
690             # convert into integer because we can't buy
691             actions = actions // self.closings
692             # round down actions to the nearest multiple
693             actions = np.where(actions >= 0,
694                 (actions // self.shares_increment) * self.s
695                 * self.shares_increment,
696             )
697             else:
698                 actions = np.where(actions > 0, actions
699
699             # clip actions so we can't sell more assets than we have
700             actions = np.minimum(actions, -np.array(self.
701
701             # deal with turbulence
702             actions = np.array(actions, dtype=np.int32)
703
703             # convert into integer because we can't buy
704             actions = actions // self.closings
705             # round down actions to the nearest multiple
706             actions = np.where(actions >= 0,
707                 (actions // self.shares_increment) * self.s
708                 * self.shares_increment,
709             )
710             else:
711                 actions = np.where(actions > 0, actions
712
712             # clip actions so we can't sell more assets than we have
713             actions = np.minimum(actions, -np.array(self.
714
714             # deal with turbulence
715             actions = np.array(actions, dtype=np.int32)
716
716             # convert into integer because we can't buy
717             actions = actions // self.closings
718             # round down actions to the nearest multiple
719             actions = np.where(actions >= 0,
720                 (actions // self.shares_increment) * self.s
721                 * self.shares_increment,
722             )
723             else:
724                 actions = np.where(actions > 0, actions
725
725             # clip actions so we can't sell more assets than we have
726             actions = np.minimum(actions, -np.array(self.
727
727             # deal with turbulence
728             actions = np.array(actions, dtype=np.int32)
729
729             # convert into integer because we can't buy
730             actions = actions // self.closings
731             # round down actions to the nearest multiple
732             actions = np.where(actions >= 0,
733                 (actions // self.shares_increment) * self.s
734                 * self.shares_increment,
735             )
736             else:
737                 actions = np.where(actions > 0, actions
738
738             # clip actions so we can't sell more assets than we have
739             actions = np.minimum(actions, -np.array(self.
740
740             # deal with turbulence
741             actions = np.array(actions, dtype=np.int32)
742
742             # convert into integer because we can't buy
743             actions = actions // self.closings
744             # round down actions to the nearest multiple
745             actions = np.where(actions >= 0,
746                 (actions // self.shares_increment) * self.s
747                 * self.shares_increment,
748             )
749             else:
750                 actions = np.where(actions > 0, actions
751
751             # clip actions so we can't sell more assets than we have
752             actions = np.minimum(actions, -np.array(self.
753
753             # deal with turbulence
754             actions = np.array(actions, dtype=np.int32)
755
755             # convert into integer because we can't buy
756             actions = actions // self.closings
757             # round down actions to the nearest multiple
758             actions = np.where(actions >= 0,
759                 (actions // self.shares_increment) * self.s
760                 * self.shares_increment,
761             )
762             else:
763                 actions = np.where(actions > 0, actions
764
764             # clip actions so we can't sell more assets than we have
765             actions = np.minimum(actions, -np.array(self.
766
766             # deal with turbulence
767             actions = np.array(actions, dtype=np.int32)
768
768             # convert into integer because we can't buy
769             actions = actions // self.closings
770             # round down actions to the nearest multiple
771             actions = np.where(actions >= 0,
772                 (actions // self.shares_increment) * self.s
773                 * self.shares_increment,
774             )
775             else:
776                 actions = np.where(actions > 0, actions
777
777             # clip actions so we can't sell more assets than we have
778             actions = np.minimum(actions, -np.array(self.
779
779             # deal with turbulence
780             actions = np.array(actions, dtype=np.int32)
781
781             # convert into integer because we can't buy
782             actions = actions // self.closings
783             # round down actions to the nearest multiple
784             actions = np.where(actions >= 0,
785                 (actions // self.shares_increment) * self.s
786                 * self.shares_increment,
787             )
788             else:
789                 actions = np.where(actions > 0, actions
790
790             # clip actions so we can't sell more assets than we have
791             actions = np.minimum(actions, -np.array(self.
792
792             # deal with turbulence
793             actions = np.array(actions, dtype=np.int32)
794
794             # convert into integer because we can't buy
795             actions = actions // self.closings
796             # round down actions to the nearest multiple
797             actions = np.where(actions >= 0,
798                 (actions // self.shares_increment) * self.s
799                 * self.shares_increment,
800             )
801             else:
802                 actions = np.where(actions > 0, actions
803
803             # clip actions so we can't sell more assets than we have
804             actions = np.minimum(actions, -np.array(self.
805
805             # deal with turbulence
806             actions = np.array(actions, dtype=np.int32)
807
807             # convert into integer because we can't buy
808             actions = actions // self.closings
809             # round down actions to the nearest multiple
810             actions = np.where(actions >= 0,
811                 (actions // self.shares_increment) * self.s
812                 * self.shares_increment,
813             )
814             else:
815                 actions = np.where(actions > 0, actions
816
816             # clip actions so we can't sell more assets than we have
8
```